

## PENGEMBANGAN SISTEM INVENTORI PERUSAHAAN MENGUNAKAN METODE USECASE DRIVEN

Irwan Tanu Kusnadi<sup>1</sup>, Apip Supiandi<sup>2</sup>, Weli Kusnadi<sup>3</sup>, Rina Riniawati<sup>4</sup>

<sup>1</sup>Sistem Informasi UBSI, <sup>2</sup>Ilmu Komputer UBSI, <sup>3</sup>Teknik Informatika STMIK PASIM,

<sup>4</sup>Sistem Informasi UBSI

[irwan.itk@bsi.ac.id](mailto:irwan.itk@bsi.ac.id), [apip.aup@bsi.ac.id](mailto:apip.aup@bsi.ac.id), [weli.kusnadi.pasim@gmail.com](mailto:weli.kusnadi.pasim@gmail.com),

[rina.rri@bsi.ac.id](mailto:rina.rri@bsi.ac.id)

### ABSTRAK

Dunia digital yang semakin hari semakin kuat dalam merubah perilaku dan pola pemikiran masyarakat membuat pengembang dan pengusaha untuk bisa mengikuti perkembangan jaman. Sistem inventori merupakan salah satu bagian dari sistem perusahaan yang kurang diperhatikan apalagi inventori yang sifatnya sekali pakai namun kenyataannya barang-barang tersebut memiliki nilai ekonomis yang cukup tinggi sehingga diperlukan pengelolaan yang baik, sehingga dapat memberikan kemudahan kepada pelaku usaha dalam melakukan pendataan dan pengadaan barang inventori. Pengembangan sistem inventori yang dikembangkan dengan menggunakan metode usecase driven dapat memberikan kemudahan dalam proses pengembangan, baik pada tahap pemodelan maupun pada tahap pengkodean karena dengan menggunakan metode usecase driven pengembangan sistem dimulai berdasarkan fasilitas yang diinginkan oleh stake holder.

Kata kunci: Sistem inventori, uml, usecase driven, visual basic, mysql

### I. PENDAHULUAN

Bisnis digital merupakan sebuah pola bisnis baru yang saat ini berkembang sangat pesat, dengan meningkatnya media pendukung yang memberikan kemudahan cukup signifikan bagi para pelaku bisnis dalam menjalankan usahanya, perkembangan bisnis digital memberikan pola hidup baru bagi masyarakat. Use case driven merupakan metode yang digunakan untuk merancang sebuah sistem dengan acuan usecase diagram sebagai tahap awal pembuatan sistem. Kelebihan dari metode ini adalah blueprint dari program yang akan dibuat telah dapat diprediksi dari awal tahap perancangan sistem, sehingga lebih mudah bagi programmer untuk mentranslasikan diagram ke dalam code program. Membangun sistem yang baik harus mengikuti tahapan perancangan sistem pada SDLC yaitu tahap analisa, perancangan, implementasi dan testing.

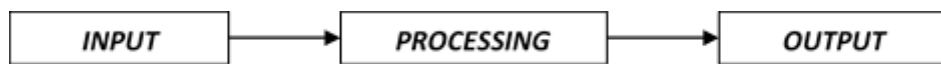
Sistem yang baik mampu memberikan efektifitas dan efisiensi yang lebih baik kepada pengguna dari sistem sebelumnya. Dalam membangun sistem basis objek, hal pertama yang harus di perhatikan adalah penentuan objek dan fasilitas yang diperlukan sistem serta memilah kebutuhan sistem tersebut menjadi kebutuhan fungsional dan non fungsional. Analisa kebutuhan sistem dari 3 aspek utama sistem yaitu bagian input, proses dan output. Seperti sebuah bangunan rancangan yang dibuat merupakan sebuah gambaran awal dari sebuah gedung dan rencana model/bentuk dari bangunan yang akan dibuat. Rancangan program merupakan gambaran awal dari program dan merupakan blueprint dari program. Use case diagram/diagram use case dalam metode usecase driven merupakan diagram awal dalam pembentukan program, use case diagram ini mirip fungsinya dengan pondasi sebuah bangunan, sehingga dengan membuat sebuah pondasi program terlebih dahulu maka programmer akan dapat lebih mudah dalam mengimplementasikan rancangan yang telah dibuat.

Sistem Inventori perusahaan biasanya menjadi bagian yang kurang diperhatikan pada perusahaan, pada kenyataannya inventori memiliki nilai ekonomis yang cukup tinggi sehingga diperlukan pengelolaan yang lebih baik, dibebberapa perusahaan sistem inventori masih menggunakan pencatatan manual dengan memcatat pada buku inventori dan membutuhkan kartu inventori untuk setiap barang namun sistem ini tentu memiliki resiko yang cukup tinggi baik dalam kesalahan pencatatan maupun dalam kehilangan serta akan sulit untuk dibuatkan laporan, baik laporan inventori barang maupun pengelolaan penanggung jawab inventori.

Latar belakang masalah yang diambil berdasarkan permasalahan yang sering terjadi pada perusahaan adalah masalah keamanan dan pengelolaan inventori yang beresiko dalam kesalahan pencatatan ataupun kehilangan baik kehilangan data maupun kehilangan barang. Masalah yang sering terjadi selanjutnya adalah masalah penyusunan laporan yang terkadang membutuhkan waktu yang cukup lama karena harus merekap secara manual.

## II. KAJIAN PUSTAKA

Sekumpulan elemen dan sub sistem yang tergabung dalam sebuah organisasi saling berinteraksi dalam mencapai tujuan tertentu baik berupa elemen program basis data maupun hardware dan software sebagai pendukung sebuah sistem. Dalam pengolahan data terdapat tiga blok dasar pengoalahan data yang harus terpenuhi yang merupakan siklus pengolahan data yaitu blok inputan data, yaitu blok yang berguna sebagai alat input(perangkat tatap muka antara komputer dengan manusia), yang kedua adalah blok pemroses yang berguna untuk proses pengolahan data yang diinput dan blok terakhir adalah blok output yang berguna sebagai penampil hasil kepada pengguna baik yang ditampilkan pada monitor maupun hasil cetak.



Gambar 1. Pengolahan Data

Merancang atau membuat sebuah pemodelan dilakukan sebelum melakukan pengkodean sesuai dengan lankah-langkah pada siklus SDLC, dengan mengikuti bagan sistem dasar yaitu membuat tiga blok dasar sistem seperti blok input, blok pemroses dan blok output, dengan melakukan perancangan model sistem yang akan dibuat dapat memberikan patokan pada saat melakukan pengkodean dan proses pembuatan program, sehingga program yang nanti terbentuk akan menjadi lebih efektif, sesuai dengan kebutuhan dan keamanan yang lebih baik.

Keberhasilan sebuah pemodelan sistem perangkat lunak dapat dilihat dari tiga unsur yaitu metode pemodelan(notasi) yaitu penggunaan notasi yang tepat, mudah dipahami dan tidak ambigu, yang kedua proses yaitu bagian pemrosesan harus sesuai dengan model yang dibuat sehingga perangkat lunak yang dibuat sesuai dengan yang diharapkan, dan yang terakhir adalah penggunaan tool dengan menggunakan *tool* yang sesuai maka perangkat lunak yang dibangun akan sesuai dengan perencanaan dan keinginan pengembang.

UML adalah bahasa pemodelan untuk perangkat lunak yang berparadigma berorientasi objek. Pemodelan (modeling) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga mudah dipelajari dan dipahami (Adi Nugroho, 2010:6).

UML merupakan salah satu bentuk bahasa yang digunakan oleh perancang sistem dalam memodelkan sebuah perangkat lunak, dengan penggunaan UML perancang(Analisis Sistem)

dapat memodelkan perangkat lunak dengan mudah karena setiap bagian digambarkan dengan objek sebagai acuan sehingga model bisa menjadi lebih mudah dipahami *stake holder*.

### III. METODE PENELITIAN

Metode penelitian yang digunakan pada penelitian ini menggunakan metode pengumpulan data dan metode pengembangan sistem yaitu:

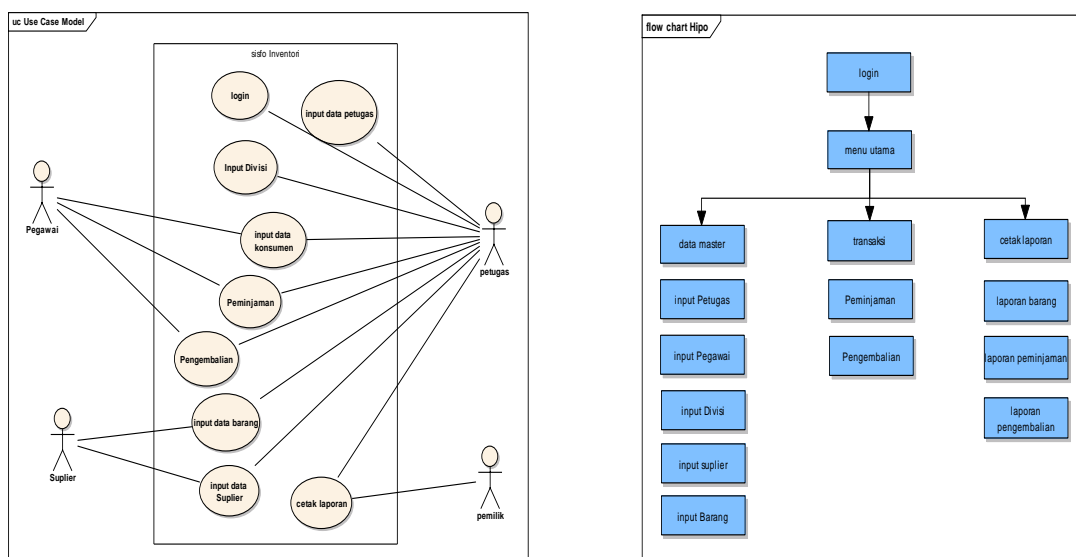
- a) Metode pengumpulan data menggunakan pendekatan pengamatan secara langsung yaitu dengan melihat langsung objek penelitian, menganalisa sistem berjalan kemudian melakukan wawancara dengan para pelaku usaha (*Stake holder*) penelitian kepustakaan dengan melihat dan menganalisa penelitian sebelumnya, sehingga didapat data-data yang dapat dipertanggungjawabkan.
- b) Metode pengembangan sistem dengan SDLC menggunakan Usecase Driven, yaitu model pengembangan sistem dengan menggunakan usecase sebagai acuan dasar dalam rancangan diagram yang lainya.

### IV. HASIL DAN PEMBAHASAN

Langkah pertama dalam membuat rancangan sistem dengan metode usecase driven adalah dengan membuat usecase sebagai acuan dalam pengembangan sistem, usecase diagram yang dibuat merupakan hasil dari analisa kebutuhan yang menggambarkan kebutuhan fungsional dan fasilitas yang ada dalam sistem yang akan dibuat.

#### Usecase Diagram

Usecase diagram memberikan gambaran umum untuk program yang akan dibuat, menjabarkan fasilitas-fasilitas yang disediakan sistem untuk membantu user dalam melakukan pekerjaannya. Penggambaran umum usecase diagram dapat di implementasikan kedalam bentuk diagram HIPO, dimana diagram HIPO merupakan bentuk nyata dari sebuah usecase.



Gambar 2. Usecase dengan Hipo

Dari penggambaran usecase diatas dapat dilihat bahwa ada kesamaan antara usecase dengan diagram hipo, sehingga dalam melakukan pengkodean seorang programmer lebih

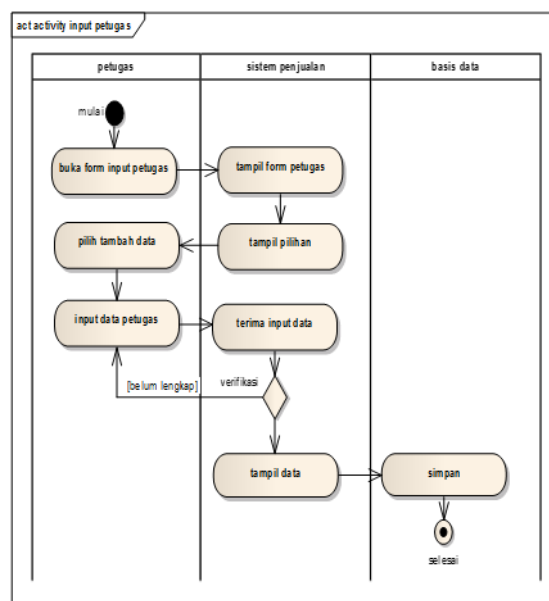
mudah dalam menjabarkan keinginan dari sistem analis dalam membuat program. Deskripsi dari usecase diagram yang dirancang adalah sebagai berikut:

Tabel 1. Deskripsi Usecase

No	Nama Usecase	Pre Conditions	Post Conditions	Goal	Keterangan
1	Login	Input Data Petugas	Menu Utama	Tampil Menu Utama	Memberikan Hak Akses Petugas
2	Input Data Petugas	-	Login	Menambah Data Petugas	Menambahkan Data Petugas(Hak Akses)
3	Input Data Barang	Login	Simpan/Edit Data Barang	Menambahkan Data Barang Ke Sistem	Mengelola Data Barang
4	Input Data Konsumen	Login	Simpan/Edit Data Konsumen	Menambahkan Data Konsumen	Mengelola Data Konsumen
5	Input Data Distributor	Login	Simpan/Edit Data Distributor	Menambahkan Data Distributor	Mengelola Data Distributor
6	Transaksi Penjualan	Login	Tambah/Simpan Tansaksi Penjualan	Catat Transaksi Penjualan	Mengelola Transaksi Penjualan Dan Perubahan Pada Stok Barang
7	Transaksi Pembelian	Login	Tambah/Simpan Transaksi Pembelian	Catat Transaksi Pembelian	Mengelola Transaksi Pembelian Dan Perubahan Pada Stok Barang
8	Cetak Laporan	Login	Tampil Laporan	Mencetak Laporan	Mencetak Laporan Untuk Pimpinan/Pemilik

### Activity Diagram

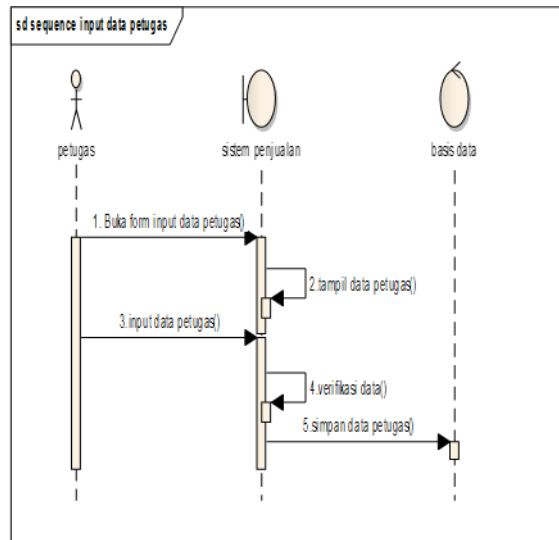
Dalam pembuatan diagram aktiviti ada beberapa teknik yang biasa dibuat namun yang paling mudah dipahami dan diimplementasi adalah dengan teknik partisi, sehingga dalam pembuatan program bisa lebih mudah karena satu diagram menggambarkan satu form program, namun kekurangannya adalah diagram yang harus dibuat menjadi lebih banyak.



Gambar 3. Diagram Activity

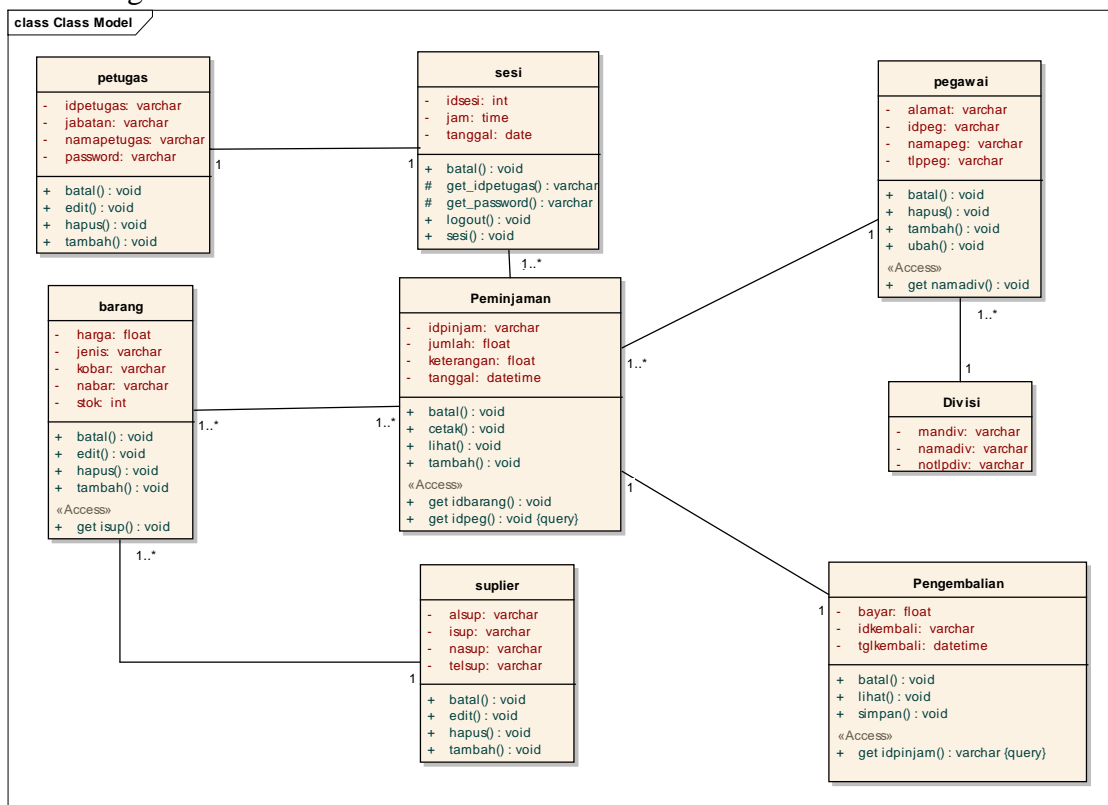
### Sequence Diagram

sequence diagram digambarkan berdasarkan diagram aktivitas dan setiap diagram aktivitas, sequence diagram menjelaskan lebih rinci alur dari aktivitas pengguna saat berinteraksi dengan sistem, sehingga programmer ataupun konsumen dapat mengetahui pasti dimana proses yang dilakukan pengguna akan di eksekusi.



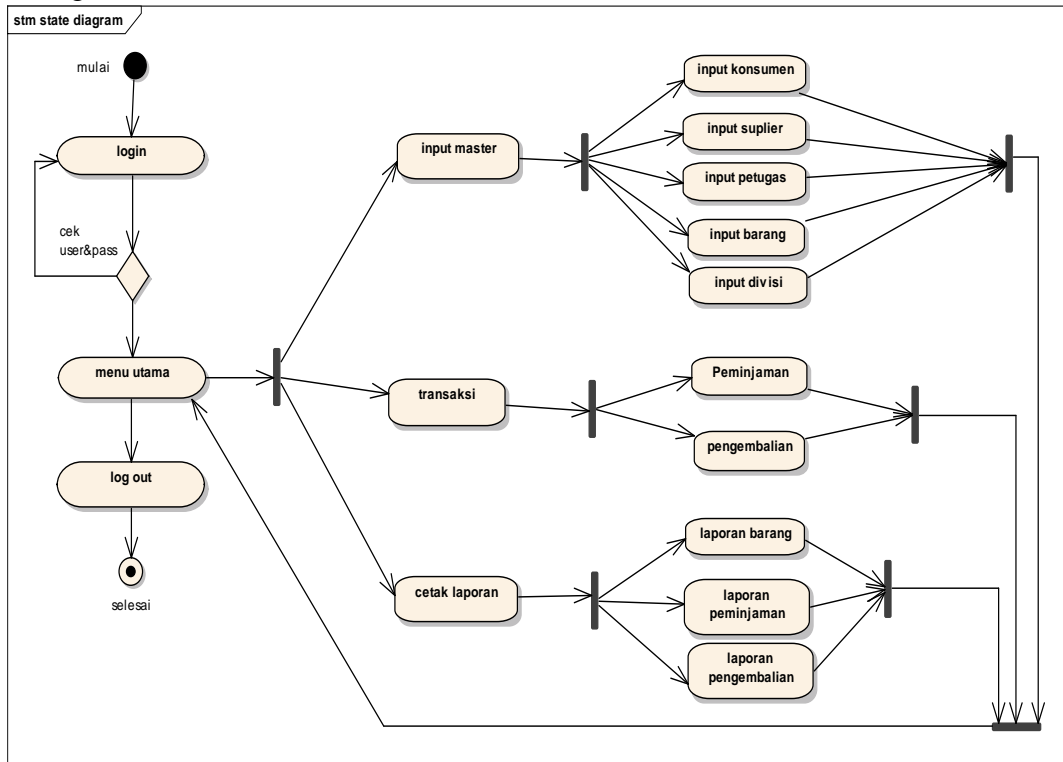
Gambar 4. Sequence Diagram

### Class Diagram



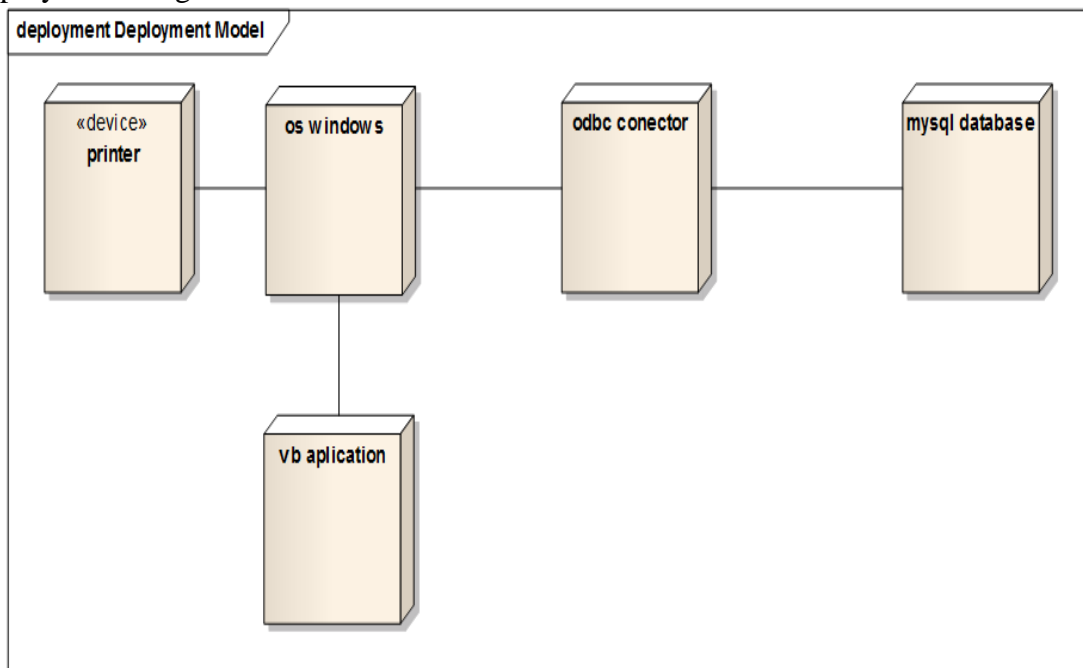
Gambar 4. Class Diagram

### State Diagram



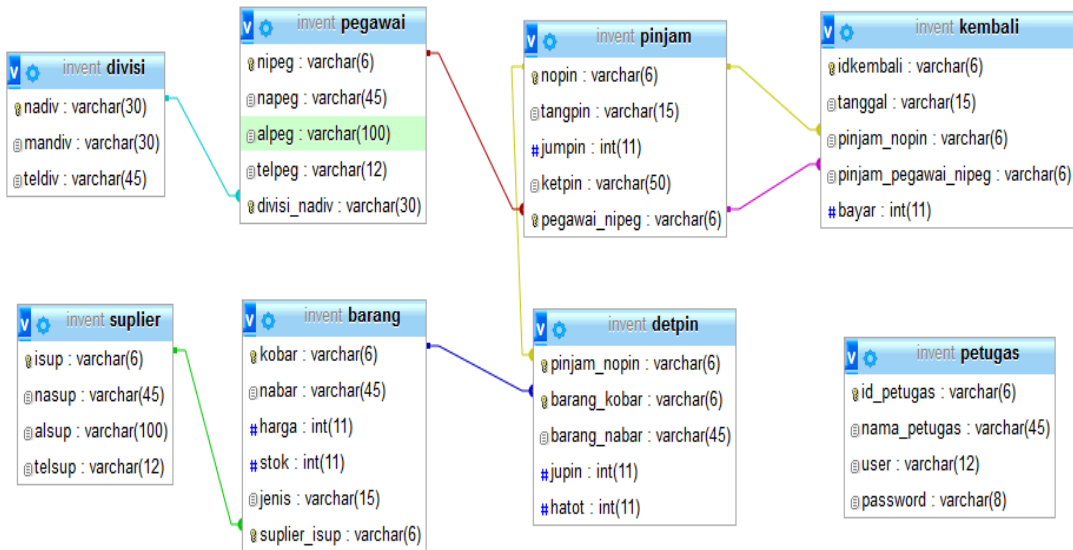
Gambar 5. State Diagram

### Deployment Diagram



Gambar 6. Deployment Diagram

## Rancangan Basis Data



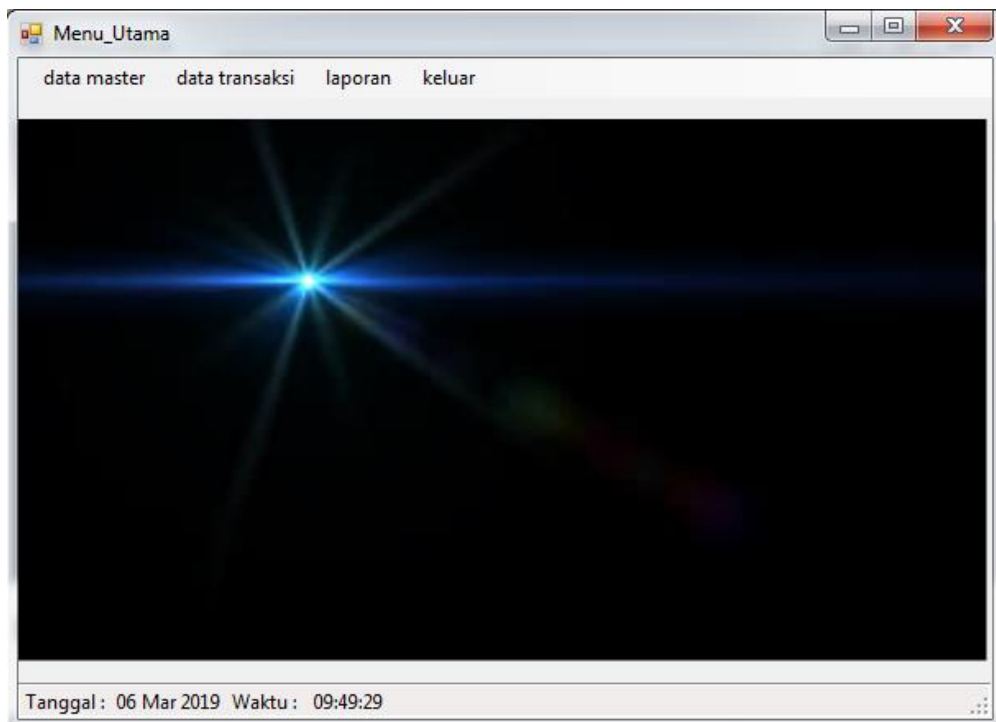
Gambar 7. Rancangan Basis Data

## Tampilan Program

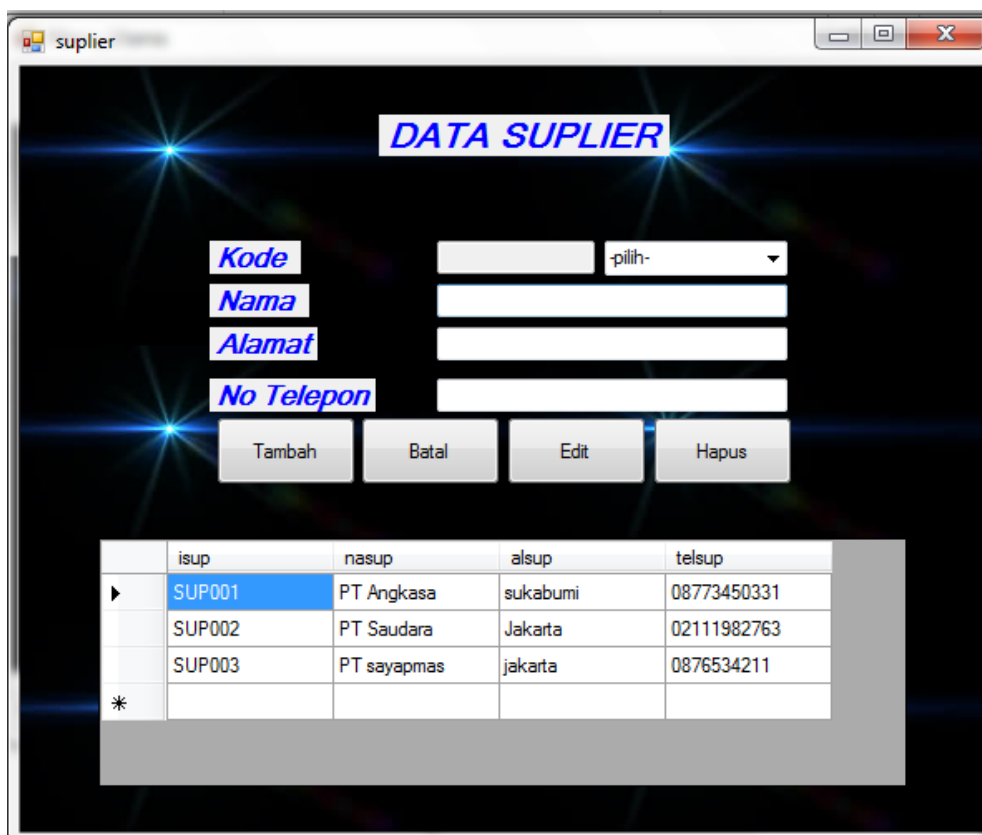
Tampilan program yang dihasilkan dari model rancangan diatas adalah sebagai berikut:



Gambar 8. Tampilan Login

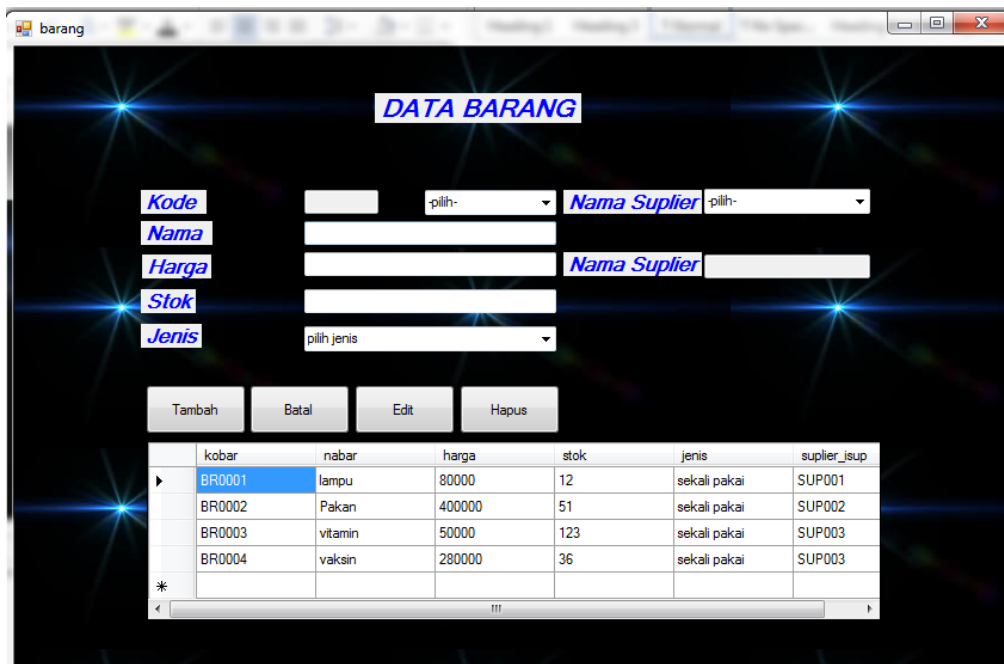


Gambar 9. Tampilan Menu Utama

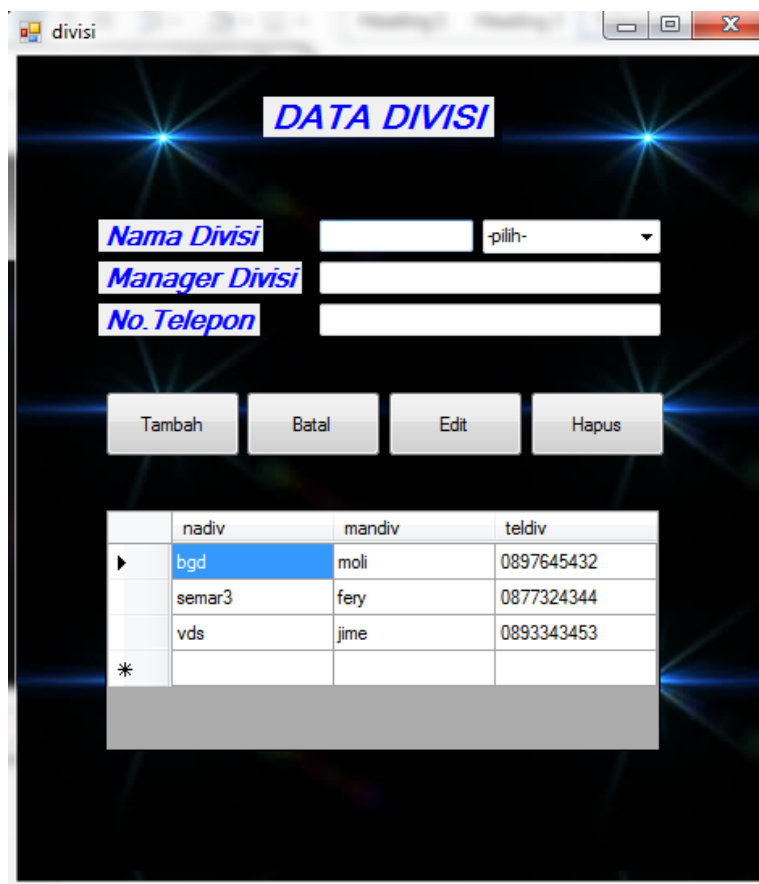


Gambar 10. Tampilan Data Suplier





Gambar 11. Tampilan Data Barang



Gambar 12. Tampilan Data Divisi

The screenshot shows a web application window titled "pegawai". The main heading is "DATA PEGAWAI". Below the heading, there are several input fields and dropdown menus for data entry:

- Kode**: A text input field.
- Nama Divisi**: A dropdown menu with "-pilih-" as the selected option.
- Nama**: A text input field.
- Alamat**: A text input field.
- No. Telepon**: A text input field.
- No Telepon**: A text input field.

Below the input fields are four buttons: "Tambah", "Batal", "Edit", and "Hapus". At the bottom of the form is a table with the following data:

	nipeg	napeg	alpeg	telpeg	divisi_nadiv
▶	087463	yati	sukabumi	08576325423	vds
	089237	jajang	sukabumi	0986543245	bgd
*					

Gambar 12. Tampilan Data Pegawai

The screenshot shows a web application window titled "petugas". The main heading is "DATA PETUGAS". Below the heading, there are several input fields and a dropdown menu for data entry:

- Kode**: A text input field.
- Nama**: A text input field.
- User**: A text input field.
- Password**: A text input field.

Below the input fields are four buttons: "Tambah", "Batal", "Edit", and "Hapus".

Gambar 13. Tampilan Data Petugas

Gambar 14. Tampilan Peminjaman

Gambar 15. Tampilan Pengembalian

laporanbarang

06 Mar 2019

**LAPORAN DATA BARANG**

	kobar	nabar	harga	stok	jenis	supplier_is
▶	BR0001	lampu	80000	12	sekali pakai	SUP001
	BR0002	Pakan	400000	51	sekali pakai	SUP002
	BR0003	vitamin	50000	123	sekali pakai	SUP003
	BR0004	vaksin	280000	36	sekali pakai	SUP003
*						

Print laporan

Gambar 16. Tampilan Laporan Data Barang

lappinjam

06 Mar 2019

**LAPORAN PEMINJAMAN**

	nopin	tangpin	jumpin	ketpin	pegawai_nipeg
▶	P0001	12 Mei 2018	1240000	kembali	089237
	P0002	13 Mei 2018	1160000	kembali	087463
*					

Print Laporan

Gambar 17. Tampilan Laporan Peminjaman

	idkembali	tanggal	pinjam_nopin	pinjam_pegawai_nij	bayar
▶	K00001	12 Mei 2018	P0001	089237	1080000
	K00002	13 Mei 2018	P0002	087463	1080000
*					

Print laporan

Gambar 18. Tampilan Laporan Pengembalian

## V. KESIMPULAN

Pengembangan sistem inventori barang yang memiliki nilai ekonomis yang tinggi sangat diperlukan untuk memberikan keamanan, efisiensi dalam penelolaanya, sistem inventori perusahaan yang dirancang dengan menggunakan metode usecase driven sangat mudah di implementasi dalam masa pengkodean karena setiap kebutuhan sistem telah dgambarkan dengan jelas pada tahapan pemodelan, dengan adanya sistem inventori pencatatan pengadaan barang, peminjaman barang dan pengebalian barang tercatat dengan baik serta penanggung jawab barang inventori pun tercatat sehingga dengan adanya sisten informasi inventori pada perusahaan dapat memberikan kemudahan dan keamanan kepada stake holder dalam melakukan pencatatan dan pengolahan barang inventori.

## DAFTAR PUSTAKA

- [1] Kusnadi, I.T, Supiandi, A, Kusnadi, W, Riniawati,R., “Rancang Bangun Sistem Informasi Persediaan Pakan dan Obat di PT. CPJF Farm Bantarsari”, Jurnal Informatika, STMIK PASIM Sukabumi, 2016, No.2 Vol.2.
- [2] Amble, Scott W, ”The element of UML 2.0 Style S.1”, Cambridge Unyversity Press.
- [3] Nugroho, A “Rekayasa Perangkat Lunak Berbasis Objek dengan Metode USDP”, penerbit ANDI, jogjakarta, 2010.
- [4] Rumbaugh, J., Jacobson, I., Booch, G.,”The Unified Modeling Language Referency Manual, 2<sup>nd</sup> Edition”, Addison-Wesley, 2004.
- [5] Pressman, R.,”Software Engineering A Practitioner’s Aproach, 7<sup>th</sup> Edition”, MCGraw-hill, Newyork, 2010.
- [6] Bennet, S, McRobb, S, Farmer, R,”Object Oriented Systems Analysis and Design Using UML, 3<sup>rd</sup> Edition. ; McGraw-Hill, 2006.
- [7] Sommerville, I,”Software Engineering, 8<sup>th</sup> Edition”, Addison-Wesley, 2007.