THE DESIGN AND DEVELOPMENT OF ADMINISTRATIVE WEB-BASED INFORMATION SYSTEM WITH COGS CALCULATION

Ibnu Gunawan [1], Yulia Kendengis [2], Roy Philip Stanley[3]
Informatics Department, Faculty of Industrial Technology,Universitas Kristen Petra, Jawa Timur, Surabaya
ibnu@petra.ac.id[1], yulia@petra.ac.id[2], m26407027@john.petra.ac.id[3]

Abstract
   Based on data from pusgrafin.go.id in 2009 stating that public paper consumption in 2009 may reach 6.45 million tons of paper demand in 2005 as many as 5.6 million tons, we can imagine how much waste paper generated. This statement exacerbating the condition of manual administration system usage by many corporate in our country
   in order to minimize this problem, we propose a go online office solution. we must make design and then developing an administrative information system that can be run using internet. And to make these information system as real as the manual system, we have been implementing our administrative web based information system in one company named UD Duta mulia.
   UD Duta mulia choosen because it have 3 branches and 1 headquarter in Surabaya. It makes the manual system so complex and thus it suitable for case study..
   The implementation result shows that the making of reports that were previously slow is now done quickly whenever a report is required. Inventory on each branch and headquarters can also now be controlled directly and thus decreased the amount of paper used by UD Duta mulia.
Keywords: COGS, Web, Administrative Information System,

1 INTRODUCTION

UD Duta Mulia is a battery store that has seven branches on the island of Java and Bali and is based in Surabaya. Until now, all business processes like sales, purchase, transaction verification, calculation of cost of goods sold and reporting is still done manually.

Some of the problems faced by the management of UD Duta Mulia, among others, is the difficulty in verifying the transaction, the process of data input, report purchases, sales reports and income statement. Verification of transactions at UD Duta Mulia is done by sending copies to the management department of a memorandum of transactions every day, and checked one by one. Financial statements up to now can not be generated because the data input process is running very slow with the large number of sales branches which must be verified and recorded. This also affects the stock settings on each branch, where the stock of each branch is governed by the leadership based on sales data and requests from shop staff. Sales report, which can not be produced quickly, slowing the business process.

And so we need an information system that utilizes information technology to solve the problem of verification, the transaction data collection, and reporting to simplify management in control of every branch. This paper would describe the process to make a good administrative web-based information system with cogs calculation using an igniter engine for php.

2 BACKGROUND

Our administrative web-based information system made based on uml and using Code Igniter on a platform of open source php. The following detailed explanation of what it uml. Code Igniter and a little discussion about COGS.

2.1 Unified Modeling Language

According to Dharwiyanti & Wahono[1], Unified Modeling Language (UML) is a "language" which has become the industry standard for visualizing, designing and documenting software systems. UML offers a standard for designing models of a system. By using UML you can create a model for all types of software applications, where the application can run on any hardware, operating system and any network, and written in any programming language. But because the UML also use the classes and operations in the basic concept, then he is more suited to writing software in object-oriented languages like C + +, Java, C # or VB.NET. However, UML can still be used for procedural modeling applications in VB or C

According to Kimmel[2], nine UML diagrams are divided into three groups based on their function, namely:

- Diagrams for requirements and design, consisting of seven diagrams, including the Use Case Diagram, Activity Diagram, Class Diagram, Object Diagram, Sequence Diagram, Collaboration Diagram, and State Diagrams.
- Diagram of a general software organization, comprised of one diagram, namely Package Diagram.
- Diagram for the implementation, consists of one diagram, namely Component & Deployment Diagram.

2.2 Code Igniter

CodeIgniter is a framework application development (Application Development Framework) - a tool - for people who build web sites using PHP. CodeIgniter goal is to help develop the site with more quickly than writing code from scratch, by providing a library that is widely used in applications in general, as well as a simple interface and logical structure to access the

library. The use CodeIgniter will help the developers focus more on the project by minimizing the amount of code that needs to be written for a job[3].

Advantages compared with the PHP framework CodeIgniter other is speed. In the conference Free and Open source Software (FrOSCon) in August 2008, the creator of PHP Rasmus Lerdorf mentioned that he liked CodeIgniter grounds "as it is faster, lighter and most like framework" than any other framework [4]

2.3 COGS

According to weygandt[2], cost of goods sold (COGS) is the entire cost of goods sold. The formula cost of goods sold is as follows: COGS = Beginning inventory + purchases of goods net - ending inventory The method is usually used in the calculation of COGS there are 3 of the First-In-First-Out (FIFO), Last-In-First-Out (LIFO), and Average. The method used in the UD Duta Mulia is average.

2.3.1 Average method

In this method, the cost of an average is calculated after each purchase made. Subsequent sales valued at an average cost of these.

3 SYSTEM DESIGN

Basically, our administrative web-based information system would be host on a web server, and then, the branch of UD Duta Mulia will be accessing the application through web. Thus it would decrease usage of paper on the administrative work. In order to design these system, we can begin with UML in system requirement phase.

3.1 System Requirement.

There are two UML diagrams which can show detailed user requirements, including the use case and activity diagrams. because many UML diagrams that available and there are small space available so this paper will only show some use cases and activity diagrams that are used. We begin with use case diagrams for login proses, as shown in figure 1
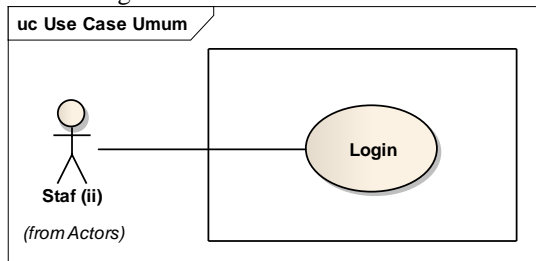

Figure 1.Use Case Diagram for login process

And then it will follow by use case diagram for branch and central process as shown in figure 2.
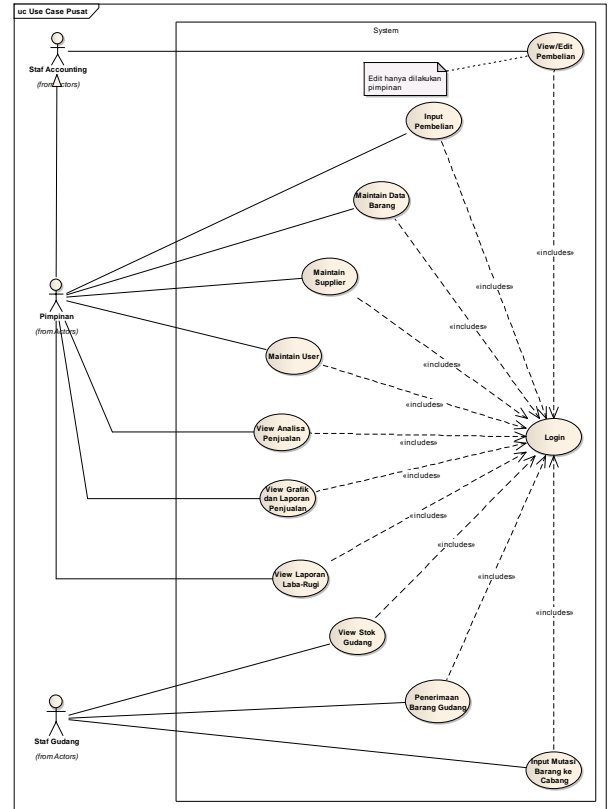
Figure 2. Use Case Diagram for central's business process

in Figure 2 we can see that the user requirement in the UD Duta Mulia Central is as follows:

- Purchasing system that consist of
  - Input / Edit Purchase system to enter or change the data of purchase
  - Including view purchase data.
- Maintain data items that consist of
  - display, add, and change the data item.
- Maintain supplier data that consist of
  - display, add, and change the data supplier.
- Viewing Sales data that consist of
  - View Sales Analysis, View Graphs and Reports Sales, Profit and Loss Report View to display the requested report.
- View Stock Vault in order to
  - view information about inventory in warehouses and in every branch.
- Acceptance of Goods Warehouse to
  - add stock every item comes from a supplier.
- Input into the Goods Movement Branch data
  - for the delivery of goods from the warehouse to the branches.

And then we can continue designing the system using activity diagrams to show who do what in the system. Figure 3 show one of the activity diagrams

supplier

idSupplier
Nama(256)
Telp(32)
alamat
kota(64)
provinsi(64)
kodePos(16)
TanggalMenjadiSupplier

pembelian

idPembelian: INTEGER
idSupplier: INTEGER
tanggalPembelian: DATE
totalPembelian: DECIMAL(15,3)
tanggalPelunasan: DATE
diterima: ENUM = belum

bonuspembelian

idBonusPembelian: INTEGER
idPembelian: INTEGER
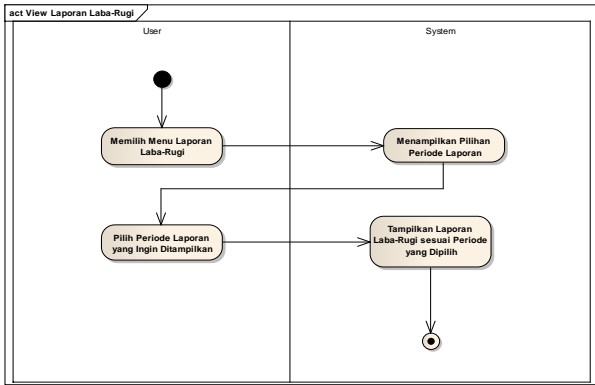Keterangan: VARCHAR(128)
Nominal: DECIMAL(15,3)

pembayaran_beli

idPembayaranBeli
idPembelian
tanggalPembayaran
metode
nominal(15,3)
keterangan(128)

pembayaran_jual

idPembayaranJual
idPenjualan
idCabang
tanggalPembayaran
metode
nominal(15,3)
keterangan(128)

penjualan

idPenjualan: INTEGER
idCabang: INTEGER
idCustomer: INTEGER
tanggalPenjualan: DATE
subtotalPenjualan: DECIMAL(15,3)
diskonPenjualan: DECIMAL(15,3)
totalPenjualan: DECIMAL(15,3)
totalPokok: DECIMAL(15,3)
tanggalPelunasan: DATE
idUser: INTEGER
status: ENUM = open

detailpembelian

idPembelian: INTEGER
idDetailPembelian: INTEGER
idItem: INTEGER
harga: DECIMAL(15,3)
jumlah: INTEGER
subtotal: DECIMAL(15,3)
diterima: INTEGER = 0

returpembelian

idReturPembelian: INTEGER
idPembelian: INTEGER
tanggalRetur: DATE
totalHarga: DECIMAL(15,3)

detailreturpembelian

iddetailReturPembelian: INTEGER
idReturPembelian: INTEGER
idItem: INTEGER
jumlah: INTEGER
harga: DECIMAL(15,3)

kategori

idKategori: INTEGER
KategoriBarang: VARCHAR(64)

returpenjualan

idPenjualan: INTEGER
idCabang: INTEGER
idReturPenjualan: INTEGER
tanggalRetur: DATE
idUser: INTEGER

detailpenjualan

idPenjualan: INTEGER
idCabang: INTEGER
idDetailPenjualan: INTEGER
idItem: INTEGER
hargaJual: DECIMAL(15,3)
jumlah: INTEGER
diskonJual: DECIMAL(15,3)
subtotalJual: DECIMAL(15,3)
hargaPokok: DECIMAL(15,3)
subtotalPokok: DECIMAL(15,3)
keterangan: TEXT

item

idItem: INTEGER
idKategori: INTEGER
Nama: VARCHAR(128)
Type: VARCHAR(128)
hargaPokok: DECIMAL(15,3)
hargaJualTerendah: DECIMAL(15,3)
hargaAverage: DECIMAL(15,3)
hargaJualTertinggi: DECIMAL(15,3)

customer

idCustomer: INTEGER
idCabang: INTEGER
Nama: VARCHAR(256)
Telp: VARCHAR(32)
alamat: TEXT
kota: VARCHAR(64)
provinsi: VARCHAR(64)
kodePos: VARCHAR(16)
TanggalMenjadiCustomer: DATE
kreditLimit: DECIMAL(15,3)

mutasi

idMutasi: INTEGER
idCabang: INTEGER
tanggalMutasi: DATE

user

idUser: INTEGER
idCabang: INTEGER
userName: VARCHAR(64)
password: VARCHAR(64)
userGroup: INTEGER

cabang

idCabang: INTEGER
nama: VARCHAR(45)
sub: VARCHAR(32)
Keterangan: TEXT

mutasi_record

idMutasi: INTEGER
idCabang: INTEGER
idMutasiRecord: INTEGER
idItem: INTEGER
jumlah: INTEGER
harga: DECIMAL(15,3)

penerimaan

idPenerimaan: INTEGER
idCabang: INTEGER
sumber: VARCHAR(32)
tanggalPenerimaan: DATE

stok_record

idStokRecord: INTEGER
tanggal: DATE
idStok: INTEGER
idCabang: INTEGER
idItem: INTEGER
masuk: INTEGER
keluar: INTEGER
sisa: INTEGER
masukrp: DECIMAL(15,3)
keluarrp: DECIMAL(15,3)
sisarp: DECIMAL(15,3)
href: VARCHAR(32)

returcabang

idReturCabang: INTEGER
idCabangAsal: INTEGER
tanggal: DATE

returcabang_record

idReturCabang: INTEGER
idCabangAsal: INTEGER
idReturCabang_record: INTEGER
idItem: INTEGER
jumlah: INTEGER
harga: DECIMAL(15,3)

detailreturpenjualan

idReturPenjualan: INTEGER
idCabang: INTEGER
iddetailReturPenjualan: INTEGER
idItem: INTEGER
jumlah: INTEGER
harga: DECIMAL(15,3)

stok

idStok: INTEGER
idCabang: INTEGER
idItem: INTEGER
jumlahItem: INTEGER

penerimaan_record

idPenerimaan: INTEGER
idCabang: INTEGER
idPenerimaanRecord: INTEGER
idItem: INTEGER
jumlah: INTEGER

Figure 5 ER-Diagram for administrative web-based information system with cogs calculation

Figure 3. activity diagram for laba – rugi report

And then we can follow it by creating a sequence diagram like shown in figure 4. To finish our design process, we can make an ER-Diagram like figure 5.
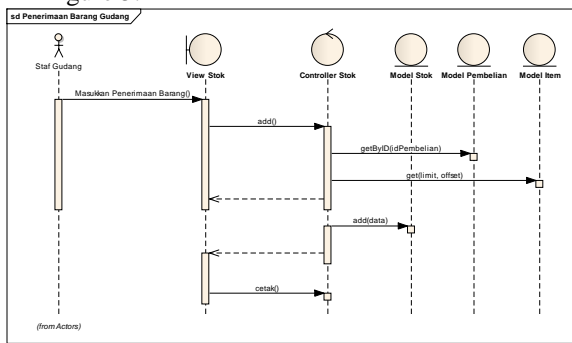

Figure 4. sequence diagram for "penerimaan barang gudang"

4 SYSTEM IMPLEMENTATION AND TESTING

In this phase we will begin to coding using code igniter. We must prepare the environment by installing XAMPPlite. Then we can follow it by activating it's service.

4.1 Database implementation

The first step is to create a database through phpMyAdmin application that can be accessed through a browser with http://localhost/phpmyadmin address. Once the database is created, which has been prepared file database directly loaded into the database via the Import menu.

To run the application, it's needs to be done through the database connection settings database configuration file located in the ". / System / application / config / database.php '. Figure 6 shown configuration that needs to be filled / modified:

```
$db['default']['username'] = "username";
$db['default']['password'] = "password";
$db['default']['database'] = "nama database";
```
Figure 6. The configuration source code

4.2 Class implementation

These class implementation make code igniter difference from other framework. In these process, simply we can follow the class diagram from UML diagram. Which is derived from sequence diagram.

There are eleven class in this system, start from cabang, customer, item, login, mutasi, pembelian. One of the class function is shown by Figure 7.

```
if($idPembelian)
{
  $data['mainview']      = 'stok/stokform';
  $data['form_action']   =
site_url('stok/add/'.$idPembelian);
  $trheader  = $this->Pembelian_model-
>getByID($idPembelian)->row();
```
Figure 7. add function source code

4.3 Application Testing

After we finished build up a web-based information system, it is now the time for testing the function to fulfilled the user requirement. Figure 8 show the login screen. Meanwhile Figure 9 show the menu screen for administrator.

Figure 8. The Login Screen

Figure 9. Menu Screen
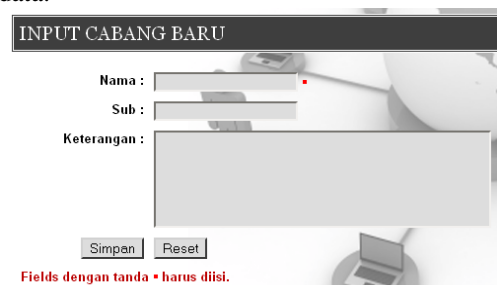
Figure 10 is the screen for inputing new branch data.

Figure 10. inputing new branch data

Figure 11 showing the user for this application

Figure 11 User list screen

This system can trap an error too, like shown in figure 12.

Figure 12 trapping user error

And as promises in abstract, that this system can make a paperless reporting, figure13 show how it happen in graphical user interface.

Figure 13 graphical reporting system

In order to close this testing, table 1 shown the final report for system testing.

Table 1. Final report for system testing

No.	Kriteria Penilai	Hasil Penilaian					
		A	B	C	D	Avg	%
1	Menu yang ada sesuai kebutuhan	5	4	5	5	4,75	95%
2	Aplikasi dapat membantu analisis	5	4	4	3	4	80%
3	Laporan yang dihasilkan	5	4	4	4	4,25	85%
4	Akurasi data	5	5	5	4	4,75	95%
5	Menu yang ada berjalan dengan baik	4	4	5	4	4,25	85%
6	Kemudahan Mengerti Aplikasi	4	3	4	3	3,5	70%
7	Kemudahan Penggunaan	4	3	4	3	3,5	70%
8	Tampilan Aplikasi	5	5	5	5	5	100%
							85%

5 CONCLUSIONS

From the design and manufacture of web-based administrative information systems is obtained several conclusions as follows:
1. The calculation of the value of inventories of goods made application in accordance with the calculation result value of inventory manually.
2. The advantages of each branch and the total of all branches can be seen
3. From the evaluation assessment conducted by the user system, it can be concluded that, overall, women rated the program is quite good with a value of 85%.

6 REFERENCES

Reference should be written using the following format:

[1] Dharwiyanti, S., & Wahono, R. S. "Pengantar Unified Modeling Language (UML)." (2006, Agustus 25). Retrieved Agustus 31, 2010, from: `http://ilmukomputer.org/2006/08/25/pengantar-uml/

[2] Weygandt, J. & Kimmel, Paul D. Accounting principles. Hobokken, NJ, Canada, 7th edition, 2005, 0471479527

[3] ExpressionEngine Development Team. "Welcome to CodeIgniter : CodeIgniter User Guide." (2006, April 3). Retrieved 09 20, 2010, from: http://codeigniter.com/user_guide/

[4] Peterson, D. "SitePoint." (2008, Agustus 29). Retrieved September 22, 2010, from: http://blogs.sitepoint.com/2008/08/29/rasmus-lerdorf-php-frameworks-think-again/