SISTEM INFORMASI *E-FORM*

PADA PT *BANK* WOORI SAUDARA

Oleh : Candra Sudirman
10512955

ABSTRACT

With the information technology advancement, data can be used and shared globally by anyone anywhere and anytime. It becomes a cooperative media for individual or group without the limitation of time and place or another factor.

With this advantage the author will design an information system in a company named Woori Saudara Bank. The objective is to minimalize paper usages such as account application form, transfer, deposit, and ATM card application form. Another problem is that customers often complain regarding a long queue taking a long time. With the information system the author will develop, the work system in the bank will be more effective. The author will develop an online web based application in a form of e-form to ease the customers in doing registration, transfer, etc. that can possibly be done anywhere and anytime, without needing to go to the bank.

Thus by the developed information system, a supporting tool is needed. The tool is Unified Modeling Language (UML). UML is a language based on graph or image to visualize, specify, develop, and document a software development system with Object-Oriented (OO) based and using descriptive research method. The system will be developed into a web based using programming language PHP for the website development.
Keywords: e-form, descriptive, website, UML

I. PENDAHULUAN

1.1 Latar Belakang Penelitian

Dewasa ini di perkembangan teknlogi ikut turut andil dalam segala peranan aktivitas manusia dimana kini segala pemanfaatannya banyak digunakan oleh banayak masyarakat dunia. Dengan ditunjang kemajuan teknologi informasi dimana data dapat digunakan dan juga dapat disebarkan secara global, siapapun dan dimanapun pengguna informasi berada. Sarana kerjasama antara pribadi atau kelompok satu dengan yang lainnya tanpa mengenal batas jarak dan waktu atau faktor lainnya.

Melalui pemanfaatannya pada sebuah perusahaan yang dimana penulis akan merancang sebuah sistem informasi pada *Bank* Woori Saudara yang beralamatkan di jalan wastukencana bandung. Dimana pada permasalahannya adalah meminimalisir penggunaan media kertas pada saat nasabah melakukan penulisan pada *form* pembukaan rekening, transfer, penyetoran, pembuatan kartu ATM, tidak jarang nasabah mengeluh akan tempat yang jauh dan antrian yang akan memakan waktu lama.

Dengan adanya sistem informasi yang akan dibangun oleh penulis yang merangkap semua kebutuhan diatas yang diantaranya penulis akan membangun suatu *web base* yang dapat digunakan secara *online* dimana aplikasi yang akan dibangun penulis berupa e-form.

Dengan demikian dari latar belakang diatas maka penulis tertarik untuk mengambil suatu judul mengenai permsalahan tersebut adalah **"SISTEM INFORMASI *E-FORM* PADA BANK WOORI SAUDARA".**

1.2 Identifikasi dan Rumusan Masalah

Berdasarkan latar belakang yang telah di uraikan, maka peneliti dapat mengidentifikasi permasalahan-permasalahan yang terjadi adalah sebagai berikut:

1. Adanya keluhan pada saat nasabah melakukan pengisian *form* pembukaan rekening yang masih menggunakan media kertas.
2. Adanya keluhan pada saat nasabah melakukan pengisian *form* transfer yang masih menggunakan media kertas.
3. Adanya keluhan pada saat nasabah melakukan pengisian *form* penyetoran yang masih menggunakan media kertas.
4. Adanya keluhan pada saat nasabah melakukan pengisian *form* pembuatan kartu yang masih menggunakan media kertas.
5. Tidak jarang nasabah mengeluh akan tempat yang jauh dan antrian yang akan memakan waktu lama.

Berdasarkan identifikasi masalah diatas, maka peneliti dapat merumuskan beberapa permasalahan yang akan di kaji lebih lanjut, yaitu sebagai berikut:

1. Bagaimana sistem belajar yang sedang berjalan di Bank Woori Saudara.
2. Bagaimana pembangunan sistem informasi *e-form* pada Bank Woori Saudara.
3. Bagaimana pengujian sistem *e-form* pada Bank Woori Saudara.
4. Bagaimana implementasi sistem *e-form* pada Bank Woori Saudara.

1.3 Maksud dan Tujuan Penelitian

Adapun maksud dari penelitian ini, yaitu untuk membangun Sistem Informasi *e-form* pada Bank Woori Saudara. Adapun tujuan dari penelitian ini, yaitu diharapkan dapat :

1. Untuk mengetahui sistem yang sedang berjalan di Bank Woori Saudara.
2. Untuk membuat perancangan sistem informasi *e-form* sehingga dapat membantu aktivitas nasabah dalam melakukan registrasi form di Bank Woori Saudara.
3. Untuk mengetahui implementasi sistem informasi *e-form* pada Bank Woori Saudara.
4. Untuk mengetahui analisis dan pengujian program di Bank Woori saudara.

1.4 Kegunaan Penelitian

Hasil dari penelitian ini diharapkan dapat memberikan manfaat diantaranya:

1. Untuk pengembangan ilmu pengetahuan, hasil penelitian ini dapat berguna bagi pengembangan ilmu pengetahuan dibidang teknologi informasi.
2. Untuk peneliti lain, dapat dijadikan sebagai bahan referensi terhadap gambaran ataupun pembuatan sistem infromasi ini.

1.5 Batasan Masalah

Batasan masalah yang diangkat adalah sebagai berikut :

1. Ruang lingkup penelitian dilakukan pada bagian CS (*Costumer Service*), *Teller*, Kabag (Kepala Bagian).

2. Sistem informasi yang diusulkan hanya dapat mengelola permohonan pembukaan rekening, permohonan transfer, permohonan penyetoran, dan permohonan pembuatan kartu.
3. Sistem hanya digunakan di bagian external perusahaan, tidak meliputi internal perusahaan.

II. KAJIAN PUSTAKA

2.1 Pengetian Sistem

Sistem dapat didefinisikan sebagai kumpulan dari komponen yang saling berhubungan satu dengan yang lainnya membentuk satu kesatuan untuk mencapai tujuan tertentu (Jogiyanto 2010).

2.2 Pengertian Informasi

Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berguna dan nyata atau berupa nilai yang dapat dipahami dalam keputusan sekarang maupun yang akan datang (Al-Bahra 2007).

2.3 Pengertian Sistem Informasi

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi organisasi yang bersifat manajerial dalam kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan – laporan yang diperlukan (Tata Sutabri 2012).

2.4 Pengertian Formulir

Formulir dapat definisikan sebagai secarik kertas atau media yang memiliki ruang untuk diisi dengan berbagai informasi sebagai dasar pencatatan transaksi atau aktifitas ekonomi suatu unit organissasi. Dapat disimpulkan bahwa formulir adalah secarik kertas yang memiliki ruang yang terformat dan diisi oleh penggunanya (Puspitawati 2011).

2.5 Pengertian E-Form

Elektronik Formulir atau Dokumen elektronik adalah konten media elektronik (selain program komputer atau file sistem) yang dimaksudkan untuk digunakan dalam bentuk elektronik atau hasil cetak. Awalnya setiap data komputer dianggap sebagai sesuau yang bersifat internal dimana data akhir selalui menggunakan media kertas. Namu dengan berkembangnya jaman dan berkembangnya teknologi komputer yang memudahkan dan mendistribusikan dokumen dokumen ke media elektronik yang kemudian dicetak sebagai hasil akhirnya (Puspitawati 2011).

III. OBJEK DAN METODE PENELITIAN

3.1 Objek dan Metode Penelitian

Penelitian yang dilakukan di PT. *Bank* Woori Saudara Indonesia 1906, Tbk KC Wastukencana Bandung, penelitian bertujuan untuk membangun sebuah sistem informasi yang dapat mempermudah aktivitas nasabah pada umumnya yang bersifat registrasi pada pembukaan rekening, transfer, penyetoran, pembuatan kartu ATM Dengan aktivitas tersebut yang melipatkan nasabah, *Customer service*, teller dan kabag. Dimana pada prosesnya nasabah tidak perlu datang langsung ke *bank* untuk mengisi formulir registrasi.

3.2 Metode Penelitian

Metode penelitian yang penulis gunakan yakni metode deskriptif yang tidak hanya terbatas pada permasalahan pengumpulan dan penyusunan data, tapi juga meliputi analisis dan interpretasi tentang arti data tersebut.

Desain penelitian adalah rencana dan struktur penyelidikan yang dibuat sedemikian rupa untuk memperoleh jawaban atas pertanyaan-pertanyaan pada penelitian (Husein Umar 2007). Desain penelitian yang penulis gunakan yakni desain penelitian *action research.* Penelitian *action research* adalah penelitian yang ditujukan untuk mengembangkan keterampilan - keterampilan baru, strategi baru atau pendekatan baru untuk memecahkan masalah dengan penerapan langsung di dunia kerja atau dunia aktual laninnya (Aries 2010).

Metode pendekatan yang penulis gunakan yakni metode pendekatan OOP (*Object Oriented Programming*) yang berfokus pada suatu objek dengan sekumpulan data yang ditetapkan dan sekumpulan operasi – operasi atau method..

Filosofi OOP menciptakan sinergi luar biasa sepanjang siklus pengembangan perangkat lunak (perencanaan, analisis, perancangan, serta implementasi) sehingga dapat diterapkan pada perancangan sistem secara umum menyangkut perangkat lunak, perangkat keras, serta sistem informasi secara keseluruhan. (Adi Nugroho 2007)

Agar sistem yang dibangun jelas kompleksitasnya dan tersusun dengan rapi penyelesainnya. Yang demikian penulis gunakan metode pengembangan *prototype* dimana metode pengembang itu sendiri menyambungkan ketidak pahaman *user* mengenai hal teknis dan memperjelas spesifikasi yang di butuhkan *user*, untuk dilakukan pengembangan pada prangkat lunak.

Dengan menggunakan model prototype *user* akan mudah membayangkan karena pada prosesnya sebagai pihak pengembang akan melakukan beberapa tahap diantaranya tahap seperti pada gambar 3.1.

Pada pengujian software ini penulis menggunakan pengujian dengan metode *black box* testing. Yang dimana *black box* testing yang berfokus pada fungsionalitas perangkat lunak sehingga memungkinkan enginners untuk memperoleh set kondisi input yang sepenuhnya dan melaksanakan persyaratan fungsionalitas pada sebuah program.

Dimana *black box* sendiri merupakan suatu keadaan yang berfokus pada persyaratan fungsional perangkat lunak yang memungkinkan *engineers* untuk memperoleh set kondisi *input* yang sepenuhnya akan melaksanakan persyaratan fungsional untuk sebuah program. (Pressman 2010)

V. HASIL PENELITIAN

4.1 Gambaran Umum Sistem yang Diusulkan

Gambaran umum sistem menjelaskan bagaimana gambaran dari sistem yang akan dibangun meliputi proses dan aktivitas yang digunakan oleh *users*, aktivitas akan menjelaskan alur dari sistem bagaimana interaksi *users* melalui aktivitas yang dibangun, dimana *users* tersebut meliputi *costumer service*, nasabah, dan teller. Berikut adalah gambaran dari sistem informasi *e-form*.

1. **Nasabah**

a. Registrasi
b. Login

c. Menu CS, Meliputi :
c.1. Pembukaan Rekening
c.2. Pembuatan Kartu
d. Menu Teller, Meliputi :
d.1. Transfer Ke Sesama
d.2. Transfer Ke Bank Lain
d.3. Setor
d.4. Penarikan

2. Costumer Service
a. Login
b. Halaman Utama
c. Nasabah, Meliputi :
c.1. Data Nasabah
c.2. Kelola Akun E-Form
d. Data Transaksi, Meliputi :
d.1. Pembukaan Rekening Baru
e. Permohonan Kartu, Meliputi :
e.1. Kartu Baru
e.2. Ganti Kartu
e.3. Blokir Kartu

3. Teller
a. Login
b. Halaman Utama
c. Nasabah, Meliputi :
b.1. Daftar Nasabah
b.2. Kelola Akun E-Form
b.3. Kelola Tabungan
b.4. Kelola Deposito
d. Data Transaksi, Meliuti :
d.1. Penyetoran
d.2. Penarikan
d.3. Tramsfer BWS
d.4. Transfer Bank Lain
e. Laporan

4. Super Kabag
a. Login
b. Halaman Utama
c. Data Master, Meliputi :
c.1. Kantor Cabang
c.2. User
c.3. Tabungan
c.4. Bank
c.5. Deposito
d. Nasabah, Meliputi :
d.1. Daftar Nasabah

d.2. Kelola Akun E-Form
d.3. Kelola Tabungan
d.4. Kelola Deposito
e. Data Transaksi, Meliputi :
e.1. Pembukaan Rekening Baru
e.2. Penyetoran
e.3. Penarikan
e.4. Transfer BWS
e.5. Transfer Bank Lainnya
f. Permohonan Kartu, Meliputi :
f.1 Kartu Baru
f.2 Ganti Kartu
f.3 Blokir Kartu
g. Laporan

4.2 Rencana Pengujian Sistem

Rencana pengujian akan menggambarkan bagaimana sebuah sistem dapat diuji menurut blackbox dimana sistem akan diuji per path diuji berdasarkan validasi yang digunakan per input. Seperti pada tabel 4.1.

4.3 Implementasi Perangkat Lunak

Perangkat lunak yang mendukung dalam menjalankan sistem informasi e-form ini, sehingga dapat dijadikan sebagai standar minimum dari spesifikasi perangkat lunak yang digunakan yakni:

1. *Operation System* : Ubuntu, Windows
2. *Browser* : Chrome, Fiefox dan Chromium

4.4 Implementasi Perangkat Keras

Untuk mendukung kinerka perangkat lunak yang digunakan maka dibuthkan suatu perangkat keras guna menjankan dan membaca program tersebut dengan baik. Berikut adalah perangkat keras yang dibutuhkan:

1. Processor : Intel Core 2
2. VGA : Asus GeForce GT 710 1GB DDR3
3. Ram : 2GB Kingstone DDR3
4. Harddisk : 250GB Seagate DDR3
5. Monitor : Dell S2240M 21.5-Inch
6. Mouse : Logitech M280
7. Keyboard : Logitech K120
8. Mainboard : Asus H110M-E D3 DDR3

4.5 Implementasi Basis Data

Implementasi basis data menjelaskan menganai alur data pada database dimana database tersebut memiliki beberapa tabel sebagai penyimpan dan penghubung arus data yang telah diolah atau akan diolah.Berikut adalah alur data dari sistem informasi *e-form*

1. Tabel banks

```
DROP TABLE IF EXISTS `banks`;
CREATE TABLE `banks` ( `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
```

`name` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `created_at` timestamp NULL DEFAULT NULL, `updated_at` timestamp NULL DEFAULT NULL, PRIMARY KEY (`id`)) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

2. Tabel costumers

```
DROP TABLE IF EXISTS `customers`;
CREATE TABLE `customers` ( `id` int(10) unsigned NOT NULL AUTO_INCREMENT, `user_id` int(10) unsigned DEFAULT NULL, `office_id` int(10) unsigned DEFAULT NULL, `full_name` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
`alias_name` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `title` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `gender` enum('pria','wanita') COLLATE utf8_unicode_ci NOT NULL, `type_of_identity` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `identity_no` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `religion` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `nationality` enum('wni','wna') COLLATE utf8_unicode_ci NOT NULL, `country` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `marital_status` enum('menikah','lajang','janda','duda') COLLATE utf8_unicode_ci NOT NULL, `mother_name` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `father_name` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `created_at` timestamp NULL DEFAULT NULL, `updated_at` timestamp NULL DEFAULT NULL, `phone` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `cif` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, PRIMARY KEY (`id`), UNIQUE KEY `customers_cif_unique` (`cif`), KEY `customers_user_id_foreign` (`user_id`), KEY `office_id` (`office_id`), CONSTRAINT `customers_ibfk_1` FOREIGN KEY (`office_id`) REFERENCES `offices` (`id`) ON DELETE SET NULL ON UPDATE CASCADE,
CONSTRAINT `customers_user_id_foreign` FOREIGN KEY (`user_id`) REFERENCES `users` (`id`) ON DELETE SET NULL) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;
```

3. Tabel customer_addresses

```
DROP TABLE IF EXISTS `customer_addresses`;
CREATE TABLE `customer_addresses` (`id` int(10) unsigned NOT NULL AUTO_INCREMENT, `customer_id` int(10) unsigned NOT NULL, `type` enum('1','2') COLLATE utf8_unicode_ci NOT NULL, `address` text COLLATE utf8_unicode_ci NOT NULL, `rt` char(3) COLLATE utf8_unicode_ci NOT NULL, `rw` char(3) COLLATE utf8_unicode_ci NOT NULL, `city` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `zipcode` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `is_same` tinyint(1) NOT NULL DEFAULT '0', `created_at` timestamp NULL DEFAULT NULL, `updated_at` timestamp NULL DEFAULT NULL, PRIMARY KEY (`id`), KEY `customer_addresses_customer_id_foreign` (`customer_id`), CONSTRAINT `customer_addresses_customer_id_foreign` FOREIGN KEY (`customer_id`) REFERENCES `customers` (`id`) ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;
```

4. Tabel customer_families

```
DROP TABLE IF EXISTS `customer_families`;
CREATE TABLE `customer_families` (`id` int(10) unsigned NOT NULL AUTO_INCREMENT, `customer_id` int(10) unsigned NOT NULL, `name` varchar(255)
```

COLLATE utf8_unicode_ci NOT NULL, `phone_number` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `address` text COLLATE utf8_unicode_ci NOT NULL, `relationship` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `created_at` timestamp NULL DEFAULT NULL, `updated_at` timestamp NULL DEFAULT NULL, PRIMARY KEY (`id`), KEY `customer_families_customer_id_foreign` (`customer_id`), CONSTRAINT `customer_families_customer_id_foreign` FOREIGN KEY (`customer_id`) REFERENCES `customers` (`id`) ON DELETE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

5. Tabel customer_financials

DROP TABLE IF EXISTS `customer_financials`;
CREATE TABLE `customer_financials` (`id` int(10) unsigned NOT NULL AUTO_INCREMENT, `customer_id` int(10) unsigned NOT NULL, `source_of_fund` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `monthly_income` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `side_income` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `source_of_side_income` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `evto_year` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `funders_name` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `funders_job` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `funders_position` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `funders_company` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `company_address` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `monthly_income_funders` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `relationship` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `created_at` timestamp NULL DEFAULT NULL, `updated_at` timestamp NULL DEFAULT NULL, PRIMARY KEY (`id`), KEY `customer_financials_customer_id_foreign` (`customer_id`), CONSTRAINT`customer_financials_customer_id_foreign` FOREIGN KEY (`customer_id`) REFERENCES `customers` (`id`) ON DELETE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

6. Tabel customer_funds

DROP TABLE IF EXISTS `customer_funds`;
CREATE TABLE `customer_funds` (`id` int(10) unsigned NOT NULL AUTO_INCREMENT, `customer_id` int(10) unsigned NOT NULL, `fund_id` int(10) unsigned DEFAULT NULL, `account_number` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `is_active` tinyint(1) NOT NULL DEFAULT '0', `created_at` timestamp NULL DEFAULT NULL, `updated_at` timestamp NULL DEFAULT NULL, PRIMARY KEY (`id`), KEY `customer_funds_customer_id_foreign` (`customer_id`), KEY `customer_funds_fund_id_foreign` (`fund_id`), CONSTRAINT `customer_funds_customer_id_foreign` FOREIGN KEY (`customer_id`) REFERENCES `customers` (`id`) ON DELETE CASCADE, CONSTRAINT `customer_funds_fund_id_foreign` FOREIGN KEY (`fund_id`) REFERENCES `funds` (`id`) ON DELETE SET NULL) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

7. Tabel customer_occupations

DROP TABLE IF EXISTS `customer_occupations`;

CREATE TABLE `customer_occupations` (`id` int(10) unsigned NOT NULL AUTO_INCREMENT, `customer_id` int(10) unsigned NOT NULL, `occupation` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `company_name` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `company_address` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `business` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `city` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `zipcode` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `phone_number` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `fax_no` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `position` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `working_time` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `created_at` timestamp NULL DEFAULT NULL, `updated_at` timestamp NULL DEFAULT NULL, PRIMARY KEY (`id`)) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

8. Tabel funds

DROP TABLE IF EXISTS `funds`;
CREATE TABLE `funds` (`id` int(10) unsigned NOT NULL AUTO_INCREMENT, `name` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `description` text COLLATE utf8_unicode_ci, `created_at` timestamp NULL DEFAULT NULL, `updated_at` timestamp NULL DEFAULT NULL, `facilities` text COLLATE utf8_unicode_ci, `terms` text COLLATE utf8_unicode_ci, PRIMARY KEY (`id`), UNIQUE KEY `funds_name_unique` (`name`)) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

9. Tabel new_cards

DROP TABLE IF EXISTS `new_cards`;
CREATE TABLE `new_cards` (`id` int(10) unsigned NOT NULL AUTO_INCREMENT, `account_id` int(10) unsigned NOT NULL, `register_id` int(10) unsigned NOT NULL, `request_type` enum('0','1','2') COLLATE utf8_unicode_ci NOT NULL, `request_argument` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `other_argument` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `with_name` tinyint(1) NOT NULL DEFAULT '0',`created_at` timestamp NULL DEFAULT NULL, `updated_at` timestamp NULL DEFAULT NULL, PRIMARY KEY (`id`), KEY `new_cards_account_id_foreign` (`account_id`), KEY `new_cards_register_id_foreign` (`register_id`), CONSTRAINT `new_cards_account_id_foreign` FOREIGN KEY (`account_id`) REFERENCES `customer_funds` (`id`) ON DELETE CASCADE, CONSTRAINT `new_cards_register_id_foreign` FOREIGN KEY (`register_id`) REFERENCES `registers` (`id`) ON DELETE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

10. Tabel new_funds

DROP TABLE IF EXISTS `new_funds`;
CREATE TABLE `new_funds` (`id` int(10) unsigned NOT NULL AUTO_INCREMENT, `customer_id` int(10) unsigned NOT NULL, `fund_id` int(10) unsigned NOT NULL, `argument` enum('0','1','2') COLLATE utf8_unicode_ci NOT NULL DEFAULT '0', `first_installment` bigint(20) NOT NULL, `created_at` timestamp NULL DEFAULT NULL, `updated_at` timestamp NULL DEFAULT NULL, `register_id` int(10) unsigned

DEFAULT NULL, `image_ktp` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `image_npwp` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, PRIMARY KEY (`id`), KEY `new_funds_customer_id_foreign` (`customer_id`), KEY `new_funds_fund_id_foreign` (`fund_id`), KEY `new_funds_register_id_foreign` (`register_id`), CONSTRAINT `new_funds_customer_id_foreign` FOREIGN KEY (`customer_id`) REFERENCES `customers` (`id`) ON DELETE CASCADE, CONSTRAINT `new_funds_fund_id_foreign` FOREIGN KEY (`fund_id`) REFERENCES `customer_funds` (`id`) ON DELETE CASCADE, CONSTRAINT `new_funds_register_id_foreign` FOREIGN KEY (`register_id`) REFERENCES `registers` (`id`) ON DELETE SET NULL) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

11. Tabel offices

DROP TABLE IF EXISTS `offices`;
CREATE TABLE `offices` (`id` int(10) unsigned NOT NULL AUTO_INCREMENT, `name` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `address` text COLLATE utf8_unicode_ci NOT NULL, `created_at` timestamp NULL DEFAULT NULL, `updated_at` timestamp NULL DEFAULT NULL, PRIMARY KEY (`id`), UNIQUE KEY `offices_name_unique` (`name`)) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

12. Tabel penarikans

DROP TABLE IF EXISTS `penarikans`;
CREATE TABLE `penarikans` (`id` int (10) unsigned NOT NULL AUTO_INCREMENT, `register_id` int(10) unsigned NOT NULL, `customer_fund_id` int(10) unsigned NOT NULL, `come_type` enum('0','1') COLLATE utf8_unicode_ci NOT NULL, `name` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL,`phone` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `sum` bigint(20) NOT NULL, `kabagistration` bigint(20) NOT NULL DEFAULT '0', `total` bigint(20) NOT NULL DEFAULT '0', `description` text COLLATE utf8_unicode_ci, `file` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `created_at` timestamp NULL DEFAULT NULL, `updated_at` timestamp NULL DEFAULT NULL, PRIMARY KEY (`id`), KEY `penarikans_customer_fund_id_foreign` (`customer_fund_id`), KEY `penarikans_register_id_foreign` (`register_id`), CONSTRAINT `penarikans_customer_fund_id_foreign` FOREIGN KEY (`customer_fund_id`) REFERENCES `customer_funds` (`id`) ON DELETE CASCADE, CONSTRAINT `penarikans_register_id_foreign` FOREIGN KEY (`register_id`) REFERENCES `registers` (`id`) ON DELETE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

13. Tabel receiver_transfers

DROP TABLE IF EXISTS `receiver_transfers`;
CREATE TABLE `receiver_transfers` (`id` int(10) unsigned NOT NULL AUTO_INCREMENT, `transfer_id` int(10) unsigned NOT NULL, `bank_id` int(10) unsigned DEFAULT NULL, `name` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `account_number` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `created_at` timestamp NULL DEFAULT NULL, `updated_at` timestamp NULL DEFAULT NULL, PRIMARY KEY (`id`), KEY `receiver_transfers_transfer_id_foreign`

(`transfer_id`), KEY `receiver_transfers_bank_id_foreign` (`bank_id`), CONSTRAINT `receiver_transfers_bank_id_foreign` FOREIGN KEY (`bank_id`) REFERENCES `banks` (`id`) ON DELETE SET NULL, CONSTRAINT `receiver_transfers_transfer_id_foreign` FOREIGN KEY (`transfer_id`) REFERENCES `transfers` (`id`) ON DELETE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

14. Tabel registers

DROP TABLE IF EXISTS `registers`;
CREATE TABLE `registers` (`id` int(10) unsigned NOT NULL AUTO_INCREMENT, `register_no` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `register_type` enum('tl','cs') COLLATE utf8_unicode_ci NOT NULL, `created_at` timestamp NULL DEFAULT NULL, `updated_at` timestamp NULL DEFAULT NULL, `user_id` int(10) unsigned DEFAULT NULL, `status` enum('0','1','2') COLLATE utf8_unicode_ci NOT NULL DEFAULT '0', PRIMARY KEY (`id`), UNIQUE KEY `registers_register_no_unique` (`register_no`), KEY `registers_user_id_foreign` (`user_id`), CONSTRAINT `registers_user_id_foreign` FOREIGN KEY (`user_id`) REFERENCES `users` (`id`) ON DELETE SET NULL) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

15. Tabel roles

DROP TABLE IF EXISTS `roles`;
CREATE TABLE `roles` (`id` int(10) unsigned NOT NULL AUTO_INCREMENT, `slug` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `name` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `permissions` text COLLATE utf8_unicode_ci, `created_at` timestamp NULL DEFAULT NULL, `updated_at` timestamp NULL DEFAULT NULL, PRIMARY KEY (`id`), UNIQUE KEY `roles_slug_unique` (`slug`)) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

16. Tabel role_users

DROP TABLE IF EXISTS `role_users`;
CREATE TABLE `role_users` (`user_id` int(10) unsigned NOT NULL, `role_id` int(10) unsigned NOT NULL, `created_at` timestamp NULL DEFAULT NULL, `updated_at` timestamp NULL DEFAULT NULL, PRIMARY KEY (`user_id`,`role_id`)) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

17. Tabel sender_transfers

DROP TABLE IF EXISTS `sender_transfers`;
CREATE TABLE `sender_transfers` (`id` int(10) unsigned NOT NULL AUTO_INCREMENT, `transfer_id` int(10) unsigned NOT NULL, `customer_fund_id` int(10) unsigned DEFAULT NULL, `method` enum('0','1') COLLATE utf8_unicode_ci DEFAULT NULL, `deposit_type` enum('0','1','2') COLLATE utf8_unicode_ci DEFAULT NULL, `intention` enum('0','1','2') COLLATE utf8_unicode_ci DEFAULT NULL, `other_intention` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `source_fund` enum('0','1','2') COLLATE utf8_unicode_ci DEFAULT NULL, `other_source_fund` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `name` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `phone` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL,`file` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `created_at` timestamp NULL DEFAULT NULL,

`updated_at` timestamp NULL DEFAULT NULL, PRIMARY KEY (`id`), KEY `sender_transfers_transfer_id_foreign` (`transfer_id`), KEY `sender_transfers_customer_fund_id_foreign` (`customer_fund_id`), CONSTRAINT `sender_transfers_customer_fund_id_foreign` FOREIGN KEY (`customer_fund_id`) REFERENCES `customer_funds` (`id`) ON DELETE SET NULL, CONSTRAINT `sender_transfers_transfer_id_foreign` FOREIGN KEY (`transfer_id`) REFERENCES `transfers` (`id`) ON DELETE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;

18. Tabel setorans

```
DROP TABLE IF EXISTS `setorans`;
CREATE TABLE `setorans` (`id` int(10) unsigned NOT NULL AUTO_INCREMENT,
 `register_id` int(10) unsigned NOT NULL, `customer_fund_id` int(10) unsigned NOT NULL, `come_type` enum('0','1') COLLATE utf8_unicode_ci NOT NULL, `intention` enum('0','1','2') COLLATE utf8_unicode_ci DEFAULT NULL, `other_intention` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `source_fund` enum('0','1','2') COLLATE utf8_unicode_ci DEFAULT NULL, `other_source_fund` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `name` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `phone` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `sum` bigint(20) NOT NULL, `kabagistration` bigint(20) NOT NULL DEFAULT '0', `total` bigint(20) NOT NULL DEFAULT '0', `description` text COLLATE utf8_unicode_ci, `file` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `created_at` timestamp NULL DEFAULT NULL, `updated_at` timestamp NULL DEFAULT NULL, PRIMARY KEY (`id`), KEY `setorans_customer_fund_id_foreign` (`customer_fund_id`), KEY `setorans_register_id_foreign` (`register_id`), CONSTRAINT `setorans_customer_fund_id_foreign` FOREIGN KEY (`customer_fund_id`) REFERENCES `customer_funds` (`id`) ON DELETE CASCADE, CONSTRAINT `setorans_register_id_foreign` FOREIGN KEY (`register_id`) REFERENCES `registers` (`id`) ON DELETE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;
```

19. Tabel transfers

```
DROP TABLE IF EXISTS `transfers`;
CREATE TABLE `transfers` (`id` int(10) unsigned NOT NULL AUTO_INCREMENT,
 `customer_id` int(10) unsigned NOT NULL, `register_id` int(10) unsigned NOT NULL,
 `description` text COLLATE utf8_unicode_ci, `sum` bigint(20) NOT NULL, `kabagistration` bigint(20) NOT NULL DEFAULT '0', `total` bigint(20) NOT NULL DEFAULT '0', `no_warkat` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL,
 `bank_warkat` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL, `type` enum('0','1') COLLATE utf8_unicode_ci NOT NULL DEFAULT '0', `created_at` timestamp NULL DEFAULT NULL, `updated_at` timestamp NULL DEFAULT NULL, PRIMARY KEY (`id`), KEY `transfers_customer_id_foreign` (`customer_id`), KEY `transfers_register_id_foreign` (`register_id`), CONSTRAINT `transfers_customer_id_foreign` FOREIGN KEY (`customer_id`) REFERENCES `customers` (`id`) ON DELETE CASCADE, CONSTRAINT `transfers_register_id_foreign` FOREIGN KEY (`register_id`) REFERENCES `registers` (`id`) ON DELETE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;
```

20. Tabel users

```
DROP TABLE IF EXISTS `users`;
CREATE TABLE `users` ( `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `email` varchar(255) COLLATE utf8_unicode_ci NOT NULL,  `password` varchar(255) COLLATE utf8_unicode_ci NOT NULL, `permissions` text COLLATE utf8_unicode_ci,
  `last_login` timestamp NULL DEFAULT NULL,  `first_name` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL,  `last_name` varchar(255) COLLATE utf8_unicode_ci DEFAULT NULL,  `created_at` timestamp NULL DEFAULT NULL,  `updated_at` timestamp NULL DEFAULT NULL,  `username` varchar(255) COLLATE utf8_unicode_ci NOT NULL,  `is_active` tinyint(1) NOT NULL DEFAULT '0', `remember_token` varchar(100) COLLATE utf8_unicode_ci DEFAULT NULL, `office_id` int(10) unsigned DEFAULT NULL,  PRIMARY KEY (`id`),  UNIQUE KEY `users_email_unique` (`email`),  UNIQUE KEY `users_username_unique` (`username`),  KEY `office_id` (`office_id`),  CONSTRAINT `users_ibfk_1` FOREIGN KEY (`office_id`) REFERENCES `offices` (`id`) ON DELETE SET NULL ON UPDATE CASCADE) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;
```

4.6 Implementasi Antarmuka

1. Halaman Nasabah

   a) Halaman *Home*

   Halaman *home* ini adalah halaman utama dari nasabah pada saat berhasil melakukan login. Pada halaman ini, nasabah dipermudah dengan adanya icon/menu bar untuk mengakses modul yang tersedia pada sistem. Tampilan dari dashboard dapat dilihat pada gambar 4.1.

   b) Halaman pembukaan rekening

   Melakukan pembukaan rekening bertujuan untuk nasabah melakukan pembukaan rekening sesuai dengan jenis tabungan yang dipilih nasabah dan pembukaan tabungan tersebut bisa dibuat untuk nasabah yang sudah memiliki maupun yang belum memiliki. Seperti pada gambar 4.2.

   c) Halaman pembukaan kartu

   Melakukan pembukaan kartu bertujuan untuk nasabah membuka kartu kartu ATM dimana sistem akan memperbolehkan nasabah memiliki kartu ATM jika nasabah sudah memiliki nomor rekening atau buku tabungan dapat dilihat pada gambar 4.3.

   d) Halaman transfer

   Melakukan transfer bertujuan untuk memudahkan nasabah dalam melakukan pengiriman dana dimana pada transfer tersebut memiliki dua kategori diantaranya setor ke sesama dan ke bank lainnya dapat dilihat pada gambar 4.4.

   e) Halaman penarikan

   Melakukan penarikan bertujuan untuk mengambil sejumlah dana pada nomor rekening nasabah yang bersangkutan dapat dilihat pada gambar 4.5.

V. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berikut adalah kesimpulan mengenai sistem yang dibangun adalah sebagai berikut:

Dengan sistem yang dibangun penulis akan memudahkan pengguna dalam melakukan pengisian pembukaan rekening, transfer, penyetoran tunai sehingga dapat dilakukan dimana saja dan kapan saja, selain itu nomor antrian akan secara otomatis tergenerate.

5.2. Saran

Dengan melalui saran yang dapat membangun sebuah sistem yang dibangun penulis ke arah yang lebih baik, dimana pada sistem yang akan disarankan diharapkan sistem dapat menunjang kebutuhan pada sarana:

1. Ruang lingkup penelitian yang akan dikembangkan oleh penulis selebihnya tidak hanya meliputi padabagian CS, teller dan kabag.
2. Pengelolaan pada sistem tidak hanya meliputi permohonan pembukaan rekening, permohonan transfer, permohonan 6penyetoran, dan permohonan pembuatan kartu.
3. Sistem yang digunakan dapat menunjang kebutuhan internal perusahaan seperti terknoneksinya sistem tersebut dengan internal perusahaan

VI. DAFTAR PUSTAKA

Jogiyanto, “Analisis dan Desain Sistem Informasi”, 4ed rd, Yogyakarta: Andi Offset. 2010

Al-Bahra bin Ladjamudin, “Analisis dan Desain Sistem Informasi”. 2rd ed, Graha Ilmu: Yogyakarta. 2007.

Tata Sutabri, “Analisis Sistem Informasi”. 1rd ed, Yogyakarta: Andi. 2012.

Puspitawati. “Sistem Informasi Akuntansi”. 1rd ed, Yogyakarta: Graha Ilmu. 2011

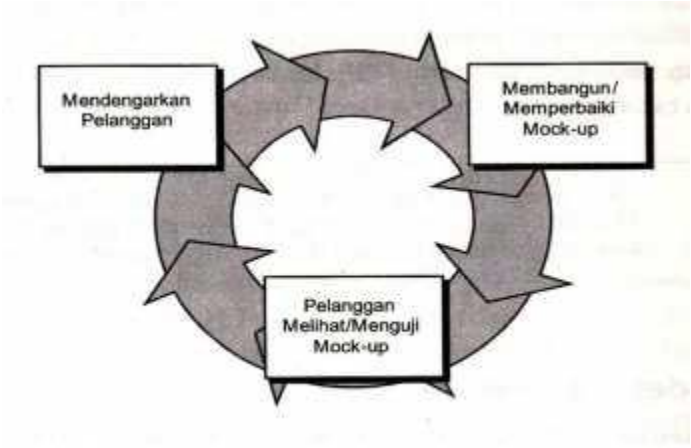
Husein Umar, "Metode Penelitian Untuk Skripsi Dan Tesis Bisnis", 1rd ed, Jakarta : PT. Raja Grafindo Persada, 2007

Aries, Erna Febru, "*Design Action Research*", 1rd ed, Yogyakarta: Aditya Media Publishing, 2010.

Adi Nugroho. “Analisis Dan Perancangan Sistem Informasi Dengan Metodologi Berorientasi Objek”, 3rd ed, Informatika, Bandung: 2007.

Pressman, “*Software Engineering A Practitioner’s Approach*”, 1rd ed, New York: McGraw-Hill. 2010

LAMPIRAN TABEL DAN GAMBAR

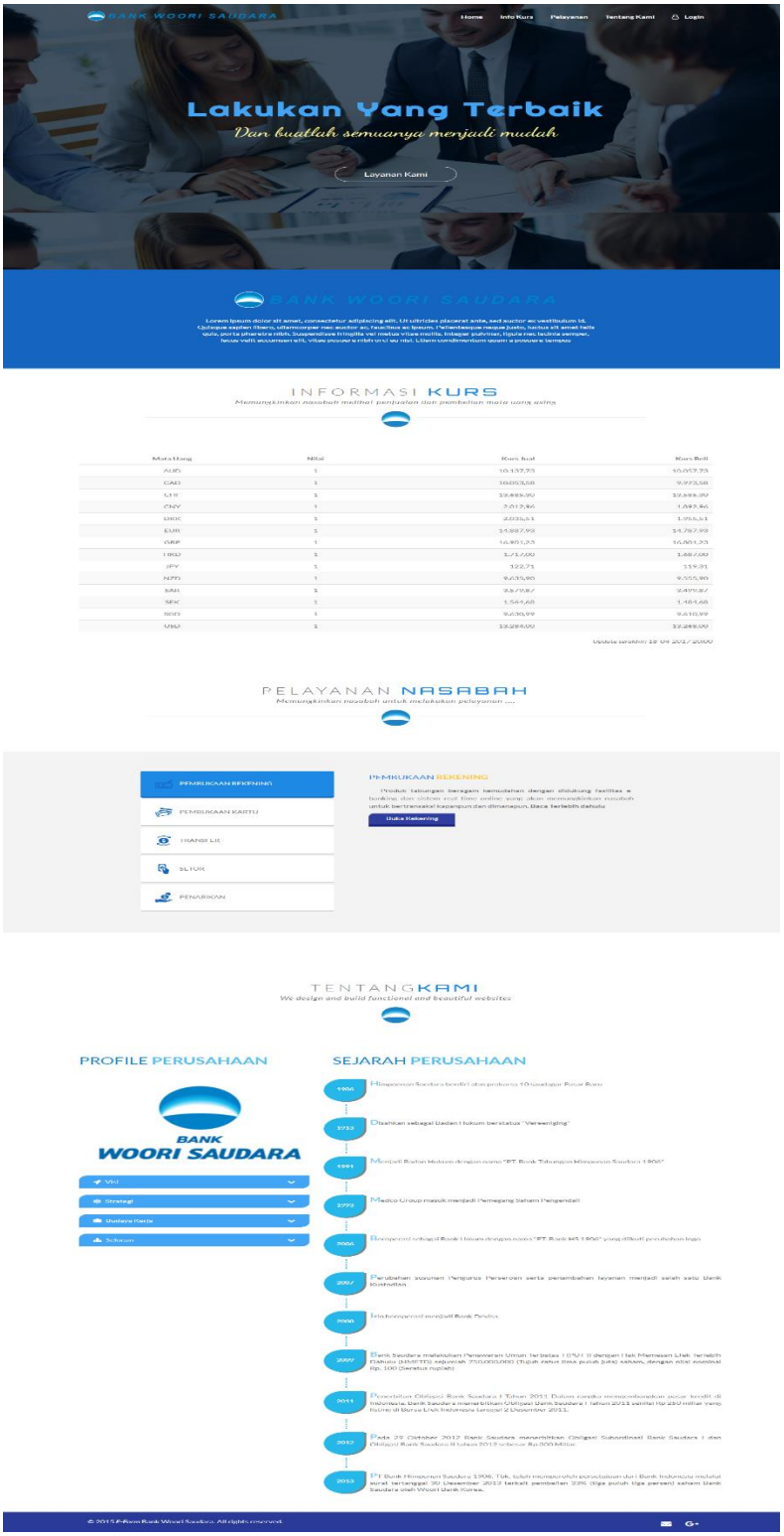


Gambar 3.1 Model Prototype
(Sumber : Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur & Berorientasi Objek)

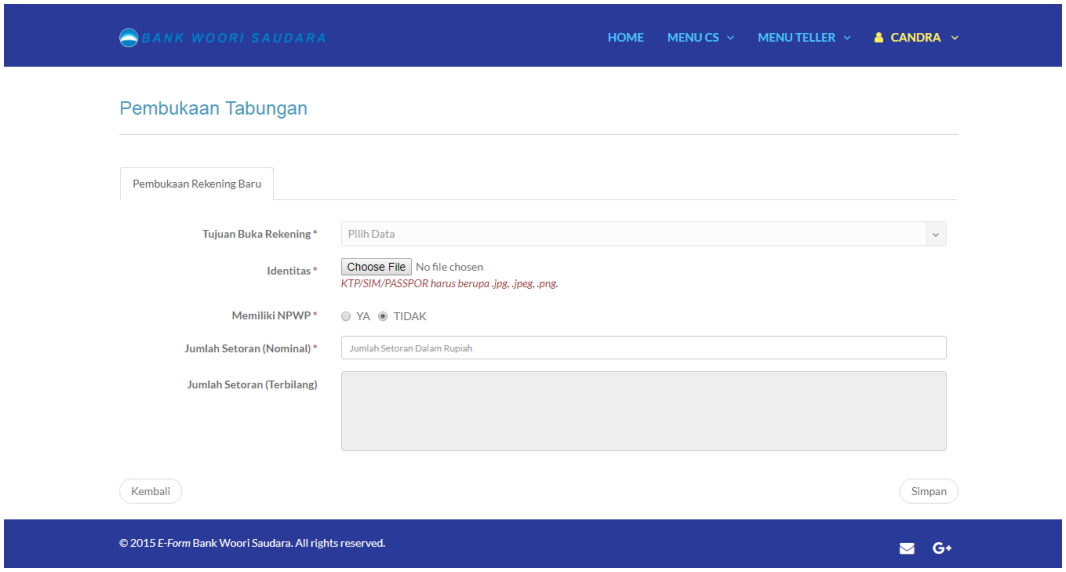
Tabel 4.1 Rencana Pengujian

Kelas Uji	Tingkat Uji	Butir Uji	Jenis Pengujian
Login	*Email*	Modul	*Blackbox Testing*
	Password	Modul	*Blackbox Testing*
Data *Master*	Olah Data Kantor Cabang	Modul	*Blackbox Testing*
	Olah Data Bank	Modul	*Blackbox Testing*
	Olah Data User	Modul	*Balckbox Testing*
	Olah Data Tabungan	Modul	*Balckbox Testing*
Proses	Kelola Pembukaan Rekening	Modul	*Blackbox Testing*
	Kelola Fasilitas Kartu	Modul	*Blackbox Testing*

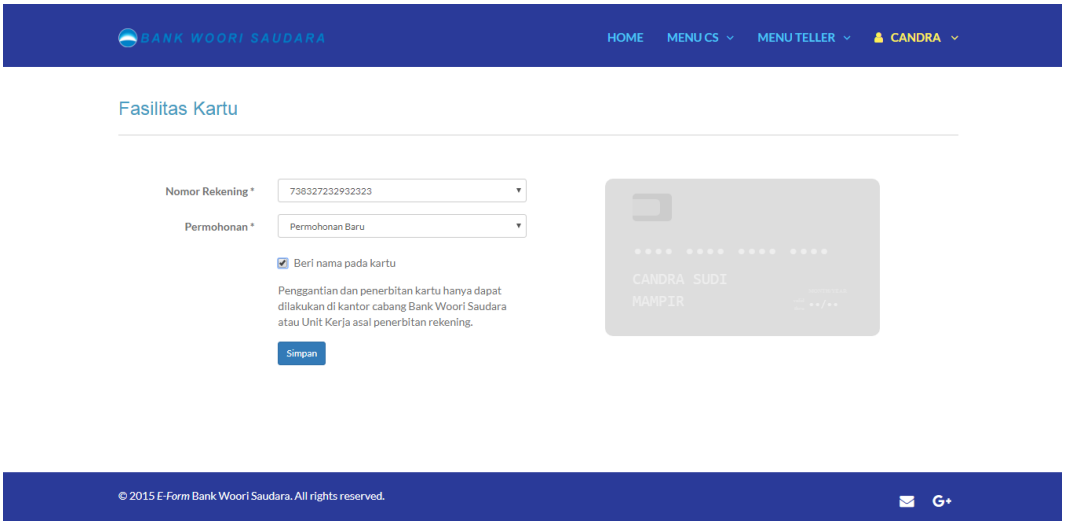
	Kelola Transfer Ke Sesama	Modul	*Blackbox Testing*
	Kelola Transfer K Bank Lain	Modul	*Blackbox Testing*
	Kelola Setor	Modul	*Blackbox Testing*
	Kelola Penarikan	Modul	*Blackbox Testing*
Output	View Data Cetak Bukti Transaksi	Modul	*Blackbox Testing*
	View Data Penyetoran	Modul	*Blackbox Testing*
	View Data Pembukaan Rekening Baru	Modul	*Blackbox Testing*
	View Data Penarikan	Modul	*Blackbox Testing*
	View Data Transfer BWS	Modul	*Blackbox Testing*
	View Data Transfer Bank Lain	Modul	*Blackbox Testing*
	View Data Permohonan Kartu Baru	Modul	*Blackbox Testing*
	View Data Permohonan Ganti Kartu	Modul	*Blackbox Testing*
	View Data Permohonan Blokir Kartu	Modul	*Blackbox Testing*
	View Data Laporan	Modul	*Blackbox Testing*



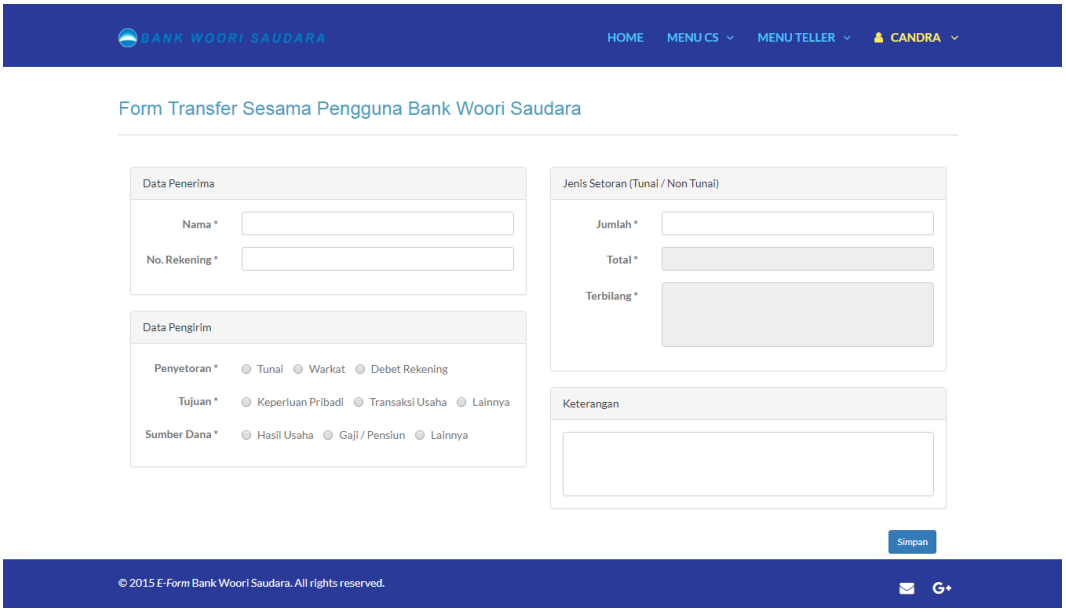
Gambar 4.1 Halaman *Home* Nasabah



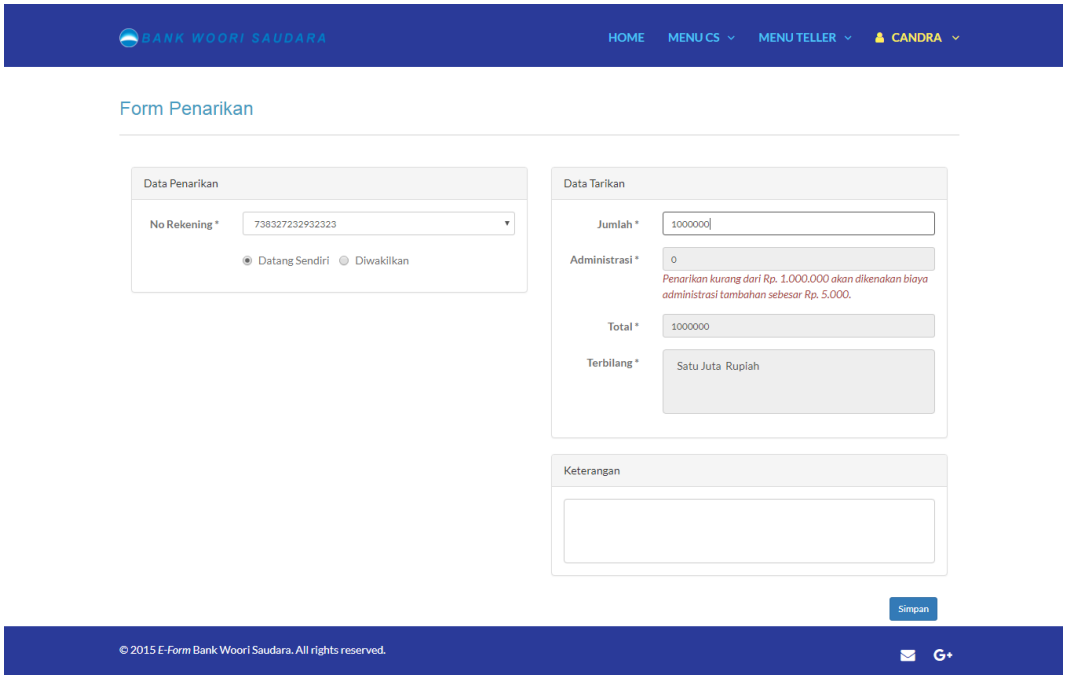
Gambar 4.2 Halaman Pembukaan Rekening



Gambar 4.3 Menampilkan Fasilitas Kartu



Gambar 4.4 Form Transfer



Gambar 4.5 Melakukan Penarikan