**PENGEMBANGAN SISTEM INVENTORY DAN PEMINJAMAN ALAT UKUR DI PT XYZ**

**TUGAS AKHIR**

Disusun oleh:

**Muhamad Sahyudi**

**3311601008**

Disusun untuk memenuhi salah satu syarat memperoleh gelar Ahli Madya Teknik Informatika



**PROGRAM STUDI TEKNIK INFORMATIKA**

**JURUSAN TEKNIK INFORMATIKA**

**POLITEKNIK NEGERI BATAM**

**BATAM**

**2018**

# 

# **HALAMAN PERNYATAAN**

Dengan ini, saya:

NIM : Muhamad Sahyudi

Nama : 3311601008

adalah mahasiswa Program Studi Teknik Informatika Politeknik Negeri Batam menyatakan bahwa Tugas Akhir dengan judul:

**PENGEMBANGAN SISTEM INVENTORY DAN PEMINJAMAN ALAT UKUR DI PT XYZ**

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 Negeri Batam untuk mempergunakan, mendistribusikan ataupun memproduksi ulang seluruh hasil Tugas Akhir ini.

Batam, 28 Desember 2018

**Muhamad Sahyudi**

**3311601008**

# **ABSTRAK**

**PENGEMBANGAN SISTEM INVENTORI DAN PEMINJAMAN ALAT UKUR DI PT XYZ**

Inventaris menjadi sangat penting untuk mencatat dan mengelola persediaan barang yang dimiliki oleh suatu perusahaan. Pencatatan barang yang di pinjam dan dikembalikan harus tercatat dengan baik dan rapi. Informasi kerusakkan yang terjadi pada alat juga harus segera dilaporkan, dan pada alat ukur kalibrasi mennjadi proses yang sangat penting demi memastikan alat tersebut digunakan dalam kondisi yang normal. Untuk itu dibutuhkan satu sistem yang bisa melakukan proses itu semua demi menjaga kinerja suatu perusahaan. Dengan memanfaatkan teknologi yang berkembang sebagai alat bantu untuk memudahkan dalam melakukan proses invetaris, peminjaman, pengembalian dan dapat melakukan kalibrasi pada alat ukur di PT XYZ maka perlu adanya pengembangan terhadap aplikasi yang sudah ada. Pengembangan sistem informasi berbasis web ini menggunakan *database postgreSQL* sebagai pendukung untuk melakukan pengolahan dan tersebut. Sistem informasi ini juga akan di kembangkan menggunakan *framework codeigniter.* untuk mendukung kerja pengembangan penulis menggunakan server apache yang ada pada *XAMPP.*

.

**Kata kunci**: inventaris, kalibrasi, *codeigniter,* dan *postgreSQL* .

# **ABSTRACT**

**PENGEMBANGAN SISTEM INVENTORI DAN PEMINJAMAN ALAT UKUR DI PT XYZ**

Inventory is very important to record and manage the supply of goods owned by a company. Recording items that are borrowed and returned must be recorded properly and neatly. Information about damage that occurs on the tool must also be reported immediately and in the measuring instruments calibration is a very important process to make sure that the tool can be used in normal condition. So that a system which can do all of those process in order to maintain the performance of a company is required. By utilize the development of technology as a tool to facilitate the process of inventorying, borrowing, returning, and calibrating the measuring instruments at PT. XYZ, then it is necessary to develop some applications that already exist into the web-based information system using the postgre SQL database to support the process. This information system will be developed by using Code Igniter framework. A developer use Apache server on XAMPP.

Keywords: inventory, CodeIgniter, and postgre SQL.

# **KATA PENGANTAR**

Dengan rahmat …………………………………………………………………..  
......……………………………………………………………….  
Secara umum, isi minimal adalah:  
1. Berisi ucapan rasa syukur kepada Tuhan atas selesainya Tugas Akhir.  
2. Ucapan terima kasih kepada nama-nama tertentu yang telah memberi bantuan  
selama Tugas Akhir.  
3. Menjelaskan kesan secara umum selama melaksanakan Tugas Akhir.  
Karena itu kami mengucapkan terima kasih yang sebesar-besarnya….....

Batam, ......... 2018

Penulis

**DAFTAR ISI**

[**HALAMAN PENGESAHAN** i](#_Toc534041426)

[**HALAMAN PERNYATAAN** ii](#_Toc534041427)

[**ABSTRAK** iii](#_Toc534041428)

[**ABSTRACT** iv](#_Toc534041429)

[**KATA PENGANTAR** v](#_Toc534041430)

[**DAFTAR ISI** vi](#_Toc534041431)

[DAFTAR TABEL ix](#_Toc534041432)

[DAFTAR GAMBAR x](#_Toc534041433)

[**BAB I PENDAHULUAN** 1](#_Toc534041434)

[**1.1** **Latar Belakang** 1](#_Toc534041435)

[**1.2** **Rumusan Masalah** 2](#_Toc534041436)

[**1.3** **Batasan Masalah** 3](#_Toc534041437)

[**1.4** **Tujuan** 3](#_Toc534041438)

[**1.5** **Manfaat** 3](#_Toc534041439)

[**1.6** **Sistematika Penulisan** 4](#_Toc534041440)

[**BAB II LANDASAN TEORI** 5](#_Toc534041441)

[**2.1** **Tinjauan Pustaka** 5](#_Toc534041442)

[**2.2** **CodeIgniter** 6](#_Toc534041443)

[**2.3** **PostgreSQL** 6](#_Toc534041444)

[**2.4** **XAMPP** 7](#_Toc534041445)

[**BAB III ANALISIS DAN PERANCANGAN** 8](#_Toc534041446)

[**3.6.1** **Deskripsi umum** 8](#_Toc534041447)

[**3.6.2** **Kategori Pengguna Sistem** 9](#_Toc534041448)

[**3.6.3** **Kebutuhan Fungsional** 9](#_Toc534041449)

[**3.6.4** **Kebutuhan Non Fungsional** 10](#_Toc534041450)

[**3.6.5** **Use case Diagram** 11](#_Toc534041451)

[**3.6.6** **Skenario Use Case** 11](#_Toc534041452)

[***3.6.1*** **Skenario Use Case Login** 11](#_Toc534041453)

[***3.6.2*** **Skenario Use Case Tambah Data *User*** 12](#_Toc534041454)

[***3.6.3*** **Skenario Use Case Ubah Data *User*** 12](#_Toc534041455)

[***3.6.4*** **Skenario Use Case Hapus Data *User*** 13](#_Toc534041456)

[***3.6.5*** **Skenario Use Case Tambah Data *Goods*** 13](#_Toc534041457)

[***3.6.6*** **Skenario Use Case Ubah Data *Goods*** 14](#_Toc534041458)

[***3.6.7*** **Skenario Use Case Hapus Data *Goods*** 14](#_Toc534041459)

[***3.6.8*** **Skenario Use Case Tambah Data Peminjaman** 15](#_Toc534041460)

[***3.6.9*** **Skenario Use Case Ubah Data Peminjaman** 15](#_Toc534041461)

[***3.6.10*** **Skenario Use Case Hapus Data Peminjaman** 16](#_Toc534041462)

[***3.6.11*** **Skenario Use Case Tambah Data Pengembalian** 16](#_Toc534041463)

[***3.6.12*** **Skenario Use Case Ubah Data Pengembalian** 17](#_Toc534041464)

[***3.6.13*** **Skenario Use Case Menghapus Data Pengembalian** 17](#_Toc534041465)

[***3.6.14*** **Skenario Use Case *Eskpor* Laporan** 18](#_Toc534041466)

[***3.6.15*** **Skenario Use Case Tambah data Kalibrasi *Goods*** 18](#_Toc534041467)

[***3.6.16*** **Skenario Use Case *Eskpor* Laporan Kalibarasi** 19](#_Toc534041468)

[**3.6** **Sequence Diagram** 19](#_Toc534041469)

[**3.8.1** **Sequence Diagram Login** 19](#_Toc534041470)

[**3.8.2** **Sequence Diagram Tambah Data *User*** 20](#_Toc534041471)

[**3.8.3** **Sequence Diagram Ubah Data *User*** 20](#_Toc534041472)

[**3.8.4** **Sequence Diagram Hapus *User*** 21](#_Toc534041473)

[**3.8.5** **Sequence Diagram Tambah *Goods*** 21](#_Toc534041474)

[**3.8.6** **Sequence Diagram Ubah *Goods*** 22](#_Toc534041475)

[**3.8.7** **Sequence Diagram Hapus *Goods*** 22](#_Toc534041476)

[**3.8.8** **Sequence Diagram Tambah Peminjaman** 23](#_Toc534041477)

[**3.8.9** **Sequence Diagram Ubah Peminjaman** 23](#_Toc534041478)

[**3.8.10** **Sequence Diagram Hapus Peminjaman** 24](#_Toc534041479)

[**3.8.11** **Sequence Diagram Tambah Pengembalian** 24](#_Toc534041480)

[**3.8.12** **Sequence Diagram Ubah Pengembalian** 25](#_Toc534041481)

[**3.8.13** **Sequence Diagram Hapus Pengembalian** 25](#_Toc534041482)

[**3.8.14** **Sequence Diagram Eskpor Data** 26](#_Toc534041483)

[**3.8.15** **Sequence Diagram Tambah Data Kalibrasi** 26](#_Toc534041484)

[**3.8.16** **Sequence Diagram Eskpor Data Kalibrasi** 27](#_Toc534041485)

[**3.7** **ER Diagram** 27](#_Toc534041486)

[**3.8** **Desain *User Interface*** 28](#_Toc534041487)

[**3.8.1** ***Tampilan Form login*** 28](#_Toc534041488)

[**3.8.2** ***User Interface* Tampil Data** 28](#_Toc534041489)

[**3.8.3** ***User Interface* Tambah Data** 29](#_Toc534041490)

[**3.8.4** ***User Interface* Ubah Data** 29](#_Toc534041491)

[**3.8.5** ***User Interface* Hapu Data** 30](#_Toc534041492)

# DAFTAR TABEL

[Table 0‑1. Fungsional 10](#_Toc534041493)

[Table 0‑2. Non Fungsional 10](#_Toc534041494)

[Table 0‑3. Skenario Use Case Login 12](#_Toc534041495)

[Table 0‑4.Skenario Use Case Tambah Data User 12](#_Toc534041496)

[Table 0‑5. Skenario Use Case Ubah Data User 13](#_Toc534041497)

[Table 0‑6. Skenario Use Case Hapus Data User 13](#_Toc534041498)

[Table 0‑7. Skenario Use Case Tambah Data Goods 14](#_Toc534041499)

[Table 0‑8. Skenario Use Case Ubah Data Goods 14](#_Toc534041500)

[Table 0‑9. Skenario Use Case Hapus Data Goods 15](#_Toc534041501)

[Table 0‑10. Skenario Use Case Tambah Data Peminjaman 15](#_Toc534041502)

[Table 0‑11. Skenario Use Case Ubah Data Peminjaman 16](#_Toc534041503)

[Table 0‑12. Skenario Use Case Ubah Data Peminjaman 16](#_Toc534041504)

[Table 0‑13. Skenario Use Case Tambah Data Pengembalian 17](#_Toc534041505)

[Table 0‑14. Skenario Use Case Ubah Data Pengembalian 17](#_Toc534041506)

[Table 0‑15. Skenario Use Case Menghapus Data Pengembalian 18](#_Toc534041507)

[Table 0‑16. Skenario Use Case Eskpor Laporan 18](#_Toc534041508)

[Table 0‑17. Skenario Use Case Tambah data Kalibrasi Goods 18](#_Toc534041509)

[Table 0‑18. Skenario Use Case Eskpor Laporan Kalibarasi 19](#_Toc534041510)

# DAFTAR GAMBAR

[Gambar 1 Usecase Diagram 11](#_Toc534041511)

[Gambar 2. Sequence Diagram Login 19](file:///D:\POLTEK%20KU\Agustus%202019%20WISUDA\test%20baru\Bismillah%20PA1%20Sahyudi.docx#_Toc534041512)

[Gambar 3. Sequence Diagram Tambah Data User 20](#_Toc534041513)

[Gambar 4. Sequence Diagram Ubah Data *User* 20](#_Toc534041514)

[Gambar 5. Sequence Diagram Hapus User 21](#_Toc534041515)

[Gambar 6. Sequence Diagram Tambah Goods 21](#_Toc534041516)

[Gambar 7. Sequence Diagram Ubah Goods 22](#_Toc534041517)

[Gambar 8. Sequence Diagram Hapus Goods 22](#_Toc534041518)

[Gambar 9. Sequence Diagram Tambah Peminjaman 23](#_Toc534041519)

[Gambar 10. Sequence Diagram Tambah Peminjaman 23](#_Toc534041520)

[Gambar 11. Sequence Diagram Hapus Peminjaman 24](#_Toc534041521)

[Gambar 12. Sequence Diagram Tambah Pengembalian 24](#_Toc534041522)

[Gambar 13. Sequence Diagram Ubah Pengembalian 25](#_Toc534041523)

[Gambar 14. Sequence Diagram Hapus Pengembalian 25](#_Toc534041524)

[Gambar 15. Sequence Diagram Eskpor Data 26](#_Toc534041525)

[Gambar 16. Sequence Diagram Tambah Data Kalibrasi 26](#_Toc534041526)

[Gambar 17. Sequence Diagram Eskpor Data Kalibrasi 27](#_Toc534041527)

[Gambar 18. ER Diagram 27](file:///D:\POLTEK%20KU\Agustus%202019%20WISUDA\test%20baru\Bismillah%20PA1%20Sahyudi.docx#_Toc534041528)

[Gambar 19. Tampilan Form Login 28](#_Toc534041529)

[Gambar 20. User Interface Tampil Data 28](#_Toc534041530)

[Gambar 21. User Interface Tambah Data 29](#_Toc534041531)

[Gambar 22. User Interface Ubah Data 29](#_Toc534041532)

[Gambar 23. User Interface Delete Data 30](#_Toc534041533)

# **BAB I PENDAHULUAN**

* 1. **Latar Belakang**

Perkembangan teknologi yang sangat pesat di era globalisasi ini membuat pekerjaan manusia banyak terbantu. Teknologi menjadi alat bantu manusia dalam menyelesaikan pekerjaan sehari-hari mulai dari kehidupan rumah tangga, sosial, pendidikan, dan urusan pekerjaan. Manfaat dari pemanfaatan teknologi dalam kehidupan sehari-hari sanngat banyak yang dirasakan mulai dari pekerjaan menjadi sangat mudah, waktu yang diperlukan juga relatif lebih singkat, kualitas yang dihasilkan juga menjadi lebih baik. PT XYZ adalah salah satu perusahaan yang bergerak di bidang manufaktur yang menghasilkan produk-produk untuk mendukung kegiatan eksplorasi minyak bumi.

*Gage room* adalah salah satu tempat penyimpanan yang ada di PT XYZ adapun fungsi *gage room* ialah menyimpan berbagai macam jenis khusus alat ukur yang dimana alat ukur akan digunakan dalam proses produksi untuk bisa memastikan kualitas produk yang maskimal. *Gage room* juga terdapat aktivitas peminjaman dan pengembalian alat ukur yang sudah dipinjam, dan fungsi yang tidak kalah penting pada *gage room* adalah proses kalibrasi. Kalibrasi adalah proses pemeriksaan kembali alat ukur guna memestikan alat tersebut berkerja dalam fungsi yang normal. Setiap proses kalibrasi akan memberikan *due date* di setiap alat yang telah selesai dikalibrasi. Ketika telah sampai batas waktu yang ditentukan *(due date)* alat tersebut akan di tarik dari produksi untuk melakukan kalibrasi ulang. Disistem ini kalibrasi yang kita penulis maksudkan adalah proses pemberian masa berlakukunya pemakaian alat ukur, sementara unutk proses kalibrasi alat ukurnya dilakukan terpisah.

Administrasi di *gage room* sendiri secara umumnya sudah cukup baik, karena proses pendataan alat baru yang masuk ke *gage room,* peminjaman dan pengembalian, dan pemberian masa berlaku (kalibrasi) untuk tiap alat dilakukan dengan sebauh sistem yang dibangun dari *Visual Basic.* Meskipun secara umum proses pecatatan sudah dilakukan oleh sistem akan tetapi masih banyak kendala yang ditemukan ketika praktik langsung. Minsalnya pada proses pendataan alat baru sering terjadi kesalahan akibat dari banyaknya data yang harus di *input* untuk satu data alat ukur. Tidak terdapatnya informasi untuk menampilkan alat yang harus dikalibrasi setelah sampai masanya untuk memisahkan pendataan alat yang harus dikalibrasi. Pendataan saat ini dilakukan dengan mengeskpor data dari sistem yang ada ke file *spreadsheet* baru kemudian data tersebut dipilah secara manual oleh petugas *gage room*. Setelah data dipilah secara manual berdasarkan tempat (agar lebih mudah mendata alat apa saja yang ada di satu tempat) peminjaman baru karyawan yang bertugas pergi ke tempat-tempat yang sudah di data untuk menarik alat yang sudah sampai atau masa *due date.*

Sistem yang di gunakan di *Gage room* memerlukan pengembangan agar administrasi alat ukur bisa dilakukan lebih mudah, akurat dan waktu pengerjaan yang singkat agar lebih efektif dan efisien. Adapun untuk melakukan pengembangan pada sistem yang sudah, proyek ini dilakukan oleh penulis sebagai peserta magang di PT XYZ. Adapun tujuan dari pengembangan sistem ini ialah untuk mempermudah pihak perusahaan dalam melakukan proses inventaris, peminjaman, dan kalibrasi pada alat ukur. Agar pengelolaan alat di *gage room* menjadi lebih rapi dari sisi administrasi dan efektif serta efisien dari sisi pelaksanaannya.

* 1. **Rumusan Masalah**

Berdasarkan masalah yang diuraikan diatas, maka rumusan masalah dari proyek akhir ini adalah sebagai berikut :

1. Bagaimana mengembang sistem informasi atas inventaris, peminjaman dan pengembalian alat ukur menjadi terkontrol ?
2. Bagaimana pemberian masa berlaku pada alat ukur yang sudah lulus dikalibrasi ?
3. Bagaimana informasi alat ukur yang telah sampai masa *due date* bisa ditampilkan ?
4. Bagaimana pengolahan dan laporan setiap aktivitas menjadi konsisten ?
5. Bagaimana *eskpor* data inventaris alat ukur ke *spreadsheet* ?
   1. **Batasan Masalah**

Adapun batasan masalah dalam pengembangan ini ialah sebagai berikut :

1. Pengembangan pada sistem ini bertujuan untuk merapikan administrasi pada setiap aktivitas di *gage room.*
2. Sistem informasi dibangun berbasis web menggunakan *framework Codeigniter* dan basis data *PostgreSQL.*
   1. **Tujuan**

Berdasarkan rumusan masalah yang telah sampaikan, adapun tujuan dari pengembangan sistem ini adalah :

1. Aplikasi akan dikembangkan menggunakan konsep data terpusat.
2. Setiap aktivitas yang terjadi akan dicatat kedalam sistem.
3. Porses pemeberian masa pada alat ukur yang telah lulus kalibrasi akan dilakukan oleh kalibrator sesuai ketentuan yang ada.
4. Sistem akan menampilkan secara otomatis alat ukur yang sudah sampai masa *due date*.
5. Sistem mampu memberikan laporan dari setiap aktivitas yang tercatat.
   1. **Manfaat**

Adapun manfaat dari pengembangan sistem ini adalah sebagai berikut :

1. Pendataan alat ukur menjadi lebih efektif dan efisien.
2. Laporan aktivitas menjadi lebih tepat dan waktu pengerjaan menjadi lebih singkat.
3. Bagi petugas kalibrasi pendataan alat ukur yang harus dikalibrasi menjadi lebih mudah tanpa harus memilah secara manual.
   1. **Sistematika Penulisan**

BAB I : PENDAHULUAN

Bab ini berisi tentang latar belakang permasalahan, perumusan masalah, Batasan masalah, tujuan, manfaat dan sistematika penulisan tugas akhir.

BAB II : LANDASAN TEORI

Bab ini berisi tentang landasan teori dan metode penyelesaian masalah yang digunakan dalam mendukung pembuatan system.

BAB III : ANALISIS DAN PERANCANGAN SISTEM

Bab ini berisi penjelesan langkah yang ditempuh dalam membangun sistem

BAB IV : IMPLEMENTASI

Bab ini berisi tentang implementasi dari percangan, tampilan, *coding,* dan uji coba.

BAB V : PENUTUP

Bab ini berisi tentang kesimpulan dan saran dari sistem yang telah dibangun.

# **BAB II** **LANDASAN TEORI**

* 1. **Tinjauan Pustaka**

Sebagai bahan pendukung dalam pembuatan proposal ini ada beberapa sumber referensi yang di jadikan untuk melakukan pengemabangan ini. Berikut penulis lampirkan referensi dari penelitian terdahulu yang penulis jadikan sebagai bahan pendukung :

1. Penelitian yang dilakukan oleh Pratama (2017) dengan judul “*Sistem Informasi Inventory Barang Berbasis Web pada CV. Kurnia Abdi”* adapun tujuan dari penelitian ini adalah Sistem Informasi *Inventory* memudahkan dalam dalam proses pembuatan laporan dan proses penyajian laporan. Sistem informasi *inventory* memudahkan dalam proses pencocokan data dari hasil pencatatan pada buku dan jumlah barang yang tersedia.
2. Penelitian yang dilakukan oleh Fatim Nugrahanti (2015) dengan judul “*Perancangan Sistem Informasi Inventory**Sparepart Mesin Fotocopy Dengan Menggunakan Visual Delphi 7”* adapun tujuann dari penelitian ini adalah merubah proses dari sistem yang lama ke sistem baru sehingga dalam pengolahan dan pengoperasian data *stock* barang menjadi lebih akurat.
3. Penelitian yang dilakukan oleh Sahar Janti dan Melan Susanti dengan judul “*Penerapan Sistem Inventory Sebagai Pemenuhan Kebutuhan Informasi Antar Bagian Berbasi Web Dengan Incremental Model”.* Adapun tujuan dari penelitian ini adalah data yang dihasilkan sistem valid dan *realtime.* Sistem juga memberikan informasi kepada *user* untuk melakukan pembelian barang setelah persedian sampai pada *stock* minimum sehingga tidak terjadi kekosongan barang tanpa harus menunggu laporan *harcopy* dari petugas Gudang.

Berangkat dari ragam referensi yang ada, penulis bermaksud melakukan pengembangan dari sistem yang sudah ada dengan judul “*Pengembangan Sistem InventoryDan Peminjaman Alat Ukur Di Pt XYZ”* adapun aplikasis ini dikembangankan berbasis *web.* Pengembangan ini diharapkan menjadi solusi bagi permasalah yang ada.

* 1. **CodeIgniter**

Framework merupakan kumpulan dari *class-class,* fungsi-fungsi atau prosesdur-prosedur tertentu yang sudah menjadi satu sehingga bisa mempermudah pemograman tanpa harus membuatnya dari awal. Sedangkan MVC merupakan sebuah konsep yang memisahkan pengembangan aplikasi berdasarkan komponen utama dalam membangun sebuah aplikasi. Ada 3 konsep yang pertama *View* yang menangani tampilan sebuah aplikasi. Kedua Model adalah bagian langsung yang terhubung dengan *database* untuk mengolah data. Dan yang ketiga ialah *Control* ialah jembatan penghubung untuk model dan view, interaksi ke model untuk mengambil data dari dari *database* dan interaksi ke *view* untuk menampilkannya.

Maka Codigniter adalah sebuah aplikasi yang berbentuk *framework* yang bersifat *open source* dengan menerapan konsep MVC *(model, view, controller)* yang digunakan untuk membangun aplikasi website dengan menggunakan PHP. Dengan begini pengembangn menjadi lebih mudah dan cepat.

* 1. **PostgreSQL**

*PostgreSQL* menurut Rofiq Yuliardi dalam bukunya yang berjudul “*Panduan Administrasi Database PostgreSQL”* merupakan sebuah *Object Relational Database Management System (ORDBMS)* yang dikembangkan oleh Berkeley Computer Science Departement. Sejak tahun 1996 *PostgreSQL* menjadi sebuah *database* yang menawarkan standar melebih standar yang ditetapkan oleh ANSI-SQL92 dan bisa memenuhi permintaan dunia *open source*  akan *server database SQL.*

*PostgreSQL* dalam kurun waktu lebih dari 15 tahun melakukan pengembangan secara aktif dan sudah terbukti rancangan arsitekturnya yang kuat, handal, integritas data, dan akurasi data yang sangat baik.

* 1. **XAMPP**

XAMPP menurut Yogi Wicaksono merupakan sebuah perangkat lunak yang befungsi untuk menjalankan *website* berbasis PHP. XAMPP sendiri merupakan software yang lengkap menurut penulis, di karenakan XAMPP sendiri sudah mendukung php, Apache sebagai *web server* dan *MySQL* sebagai DBMS. Akan tetapi penulis memilih *database* *PostgreSQL* untuk mengolah data dalam pengembangan ini. XAMPP yang didalamnya terdapat Apache penulis jadikan sebagai *web server* pada *computer local* atau juga bisa sebagai CPanel di *virtual server.*

# **BAB III ANALISIS DAN PERANCANGAN**

* 1. **Deskripsi umum**



Sistem *inventory* ini merupakan sistem yang melibatkan 3 jenis user. Adapun *user* tersebut ialah super admin, operator, dan kalibrator. Secara umum sistem yang dibuat ini berfungsi untuk mengatasi masalah inventaris dan peminjaman alat ukur pada *gage room* di PT XYZ Super admin merupakan master *user* dimana *user* ini bisa melakukan semua aktivitas pada sistem ini. Operator merupakan *user* yang hanya bisa untuk mengelola data alat ukur (*goods*), peminjaman, pengembalian dan mengeskpor data kalibrasi. Kalibrator merupakan *user* yang hanya bisa mengelola data kalibrasi.

Sistem sendiri dapat menampilkan data *user,* data alat ukur *(goods)*, peminjaman, pengembalian dan kalibrasi. Sistem juga menyediakan fitur eskpor data ke file *spreadsheet*. Sistem yang akan dikembangkan adalah berbasi *web* untuk bisa mengakses sistem ini menggunakan *web browser.*

* 1. **Kategori Pengguna Sistem**

Adapun kategori pengguna berdasarkan hak akses pada sistem ini ialah sebagai berikut.

1. Super Admin

Yaitu *user* dapat mengelola aplikasi secara keseluruhan yang ada :

* Mengelola data *user*
* Mengelola data alat ukur *(goods)*
* Mengelola data peminjaman
* Mengelola data pengembalian
* Mengelola data kalibrasi

1. Operator

* Mengelola data alat ukur *(goods)*
* Mengelola data peminjaman
* Mengelola data pengembalian
* Mengeskpor data kalibrasi

1. Kalibrator

* Mengelola data kalibrasi
  1. **Kebutuhan Fungsional**

Adapun kebutuh fungsional dari sistem ini adalah sebagai berikut.

|  |  |
| --- | --- |
| **ID fungsi** | **Keterangan** |
| F001 | Sistem menyediakan fitur *login* |
| F002 | Aplikasi bisa mengelola data user |
| F003 | Aplikasi dapat mengelola data *goods* |
| F004 | Aplikasi dapat mengelola data peminjaman |
| F005 | Aplikasi dapat mengelola data pengembalian |
| F006 | Aplikasi dapat meneskpor laporan data *goods* |
| F007 | Aplikasi dapat meneskpor laporan data peminjaman |
| F008 | Aplikasi dapat mengeskpor laporan data pengembalian |
| F009 | Aplikasi dapat mengelola data kolaborasi |
| F010 | Aplikasi dapat meneskpor laporan data kalibrasi |

Table 0‑1. Fungsional

* 1. **Kebutuhan Non Fungsional**

Adapun kebutuh non fungsional dari sistem ini adalah sebagai berikut.

|  |  |
| --- | --- |
| **ID Fungsi** | **Keterangan** |
| N001 | Aplikasi menggunakan antarmuka yang mudah untuk digunakan (*user friendly)* |
| N002 | Peminjaman dan pengembalian bisa di *record* menggunakan *barcode scanner* |
| N003 | Aplikasi dapat mengeprint barcode ID |

Table 0‑2. Non Fungsional

* 1. **Use case Diagram**



Gambar 1 Usecase Diagram

* 1. **Skenario Use Case**
     1. **Skenario Use Case Login**

|  |  |
| --- | --- |
| **Nama Use case** | **Melakukan Login** |
| **Deskripsi** | Semua jenis *user* harus login terlebih dahulu untuk masuk ke sistem |
| **Aktor** | Super Admin, Operator, dan Kalibrator |
| **Kondisi Awal** | *User* Memasukkan *username* dan *password* |
| **Kondisi Akhir** | Sistem menampilkan menu-menu sesuai dengan hak akses |
| **Skenario** | 1. *User* mengakses halaman *login* 2. Sistem akan menampilkan halaman *login* 3. *User* harus memasukkan *username* dan *password* pada *form* yang tersedia dihalaman *login* 4. sistem akan melakukan verifikasi data dari *username & password* yang telah dimasukkan 5. sistem menampilkan menu sesuai dengan hak akses |
| **Skenario Alternatif** | pada langkah ke – 4, jika tidak ditemukan data yang sama di dalam basis data, maka *user* akan kembali ke langkah 3. |

Table 0‑3. Skenario Use Case Login

* + 1. **Skenario Use Case Tambah Data *User***

|  |  |
| --- | --- |
| **Nama Use case** | **Mengelola data *user*** |
| **Deskripsi** | Super admin Menambah data *user* |
| **Aktor** | Super admin |
| **Kondisi Awal** | Sistem menampilkan *form* data *user* baru |
| **Kondisi Akhir** | data berhasil tambahkan dan tersimpan di basisdata |
| **Skenario** | 1. *User* mengakses *form* pendaftaran user baru. 2. Sistem akan menampilkan *form* pendaftaran *user* baru 3. *User* memasukkan data sesuai dengan *form* yang ditampilkan sistem 4. Sistem menyimpan data *user* |
| **Skenario Alternatif** | Apabila ada *form* yang masih kosonguser akan kembali ke langkah 3. |

Table 0‑4.Skenario Use Case Tambah Data User

* + 1. **Skenario Use Case Ubah Data *User***

|  |  |
| --- | --- |
| **Nama Use case** | **Mengelola data *user*** |
| **Deskripsi** | Super admin Mengubah data *user* |
| **Aktor** | Super admin |
| **Kondisi Awal** | Sistem menampilkan data *user* |
| **Kondisi Akhir** | Data berhasil diubah dan tersimpan di basisdata |
| **Skenario** | 1. *User* mengakses halaman data *user*. 2. Sistem akan menampilkan data *user.* 3. *User* memilih data yang akan diubah 4. Sistem akan menampilkan data sesuai dengan pilihan. 5. *User* merubah data yang sudah dipilih 6. *User* menyimpan data yang telah di rubah 7. Sistem akan memberikan pesan konfirmasi data berhasil di rubah |
| **Skenario Alternatif** | Pada langka ke 4 user bisa menggagalkan data yang sudah dipilih untuk dirubah, dan sistem akan kembali ke langkah 2. |

Table 0‑5. Skenario Use Case Ubah Data User

* + 1. **Skenario Use Case Hapus Data *User***

|  |  |
| --- | --- |
| **Nama Use case** | **Mengelola data *user*** |
| **Deskripsi** | Super admin Menghapus data *user* |
| **Aktor** | Super admin |
| **Kondisi Awal** | Sistem menampilkan *form* data *user* baru |
| **Kondisi Akhir** | data berhasil di hapus dari basis data |
| **Skenario** | 1. *User* mengakses halaman data *user*. 2. Sistem akan menampilkan data *user.* 3. *User* memilih data yang akan dihapus 4. Sistem akan memberikan pesan konfirmasi untuk menghapus data tersebut. 5. *User* mengkofirmasi pesan untuk menghapus data teserbut. 6. Sistem akan memberikan pesan data telah berhasil di hapus |
| **Skenario Alternatif** | Pada langka ke 4 user bisa menggagalkan data yang sudah dipilih untuk dirubah, dan sistem akan kembali ke langkah 2. |

Table 0‑6. Skenario Use Case Hapus Data User

* + 1. **Skenario Use Case Tambah Data *Goods***

|  |  |
| --- | --- |
| **Nama Use case** | **Mengolah data *goods*** |
| **Deskripsi** | Super admin dan operator menambah data *goods* |
| **Aktor** | Super admin dan operator |
| **Kondisi Awal** | Sistem menampilkan *form* data *goods* baru |
| **Kondisi Akhir** | Data berhasil tambahkan dan tersimpan di basisdata |
| **Skenario** | 1. *User* mengakses *form* pendaftaran *goods* baru. 2. Sistem akan menampilkan *form* pendaftaran *goods* baru 3. *User* memasukkan data sesuai dengan *form* yang ditampilkan sistem 4. Sistem menyimpan data *user* |
| **Skenario Alternatif** | Apabila ada *form* yang masih kosong *user* akan kembali ke langkah 3. |

Table 0‑7. Skenario Use Case Tambah Data Goods

* + 1. **Skenario Use Case Ubah Data *Goods***

|  |  |
| --- | --- |
| **Nama Use case** | **Mengolah data *goods*** |
| **Deskripsi** | Super admin dan operator mengubah data *goods* |
| **Aktor** | Super admin dan operator |
| **Kondisi Awal** | Sistem menampilkan data *goods* |
| **Kondisi Akhir** | Data berhasil diubah dan tersimpan di basisdata |
| **Skenario** | 1. *User* mengakses halaman data *goods*. 2. Sistem akan menampilkan data *goods.* 3. *User* memilih data yang akan diubah 4. Sistem akan menampilkan data sesuai dengan pilihan. 5. *User* merubah data yang sudah dipilih 6. *User* menyimpan data yang telah di rubah 7. Sistem akan memberikan pesan konfirmasi data berhasil di rubah |
| **Skenario Alternatif** | Pada langka ke 4 user bisa menggagalkan data yang sudah dipilih untuk dirubah, dan sistem akan kembali ke langkah 2. |

Table 0‑8. Skenario Use Case Ubah Data Goods

* + 1. **Skenario Use Case Hapus Data *Goods***

|  |  |
| --- | --- |
| **Nama Use case** | **Mengolah data *goods*** |
| **Deskripsi** | Super admin dan operator menghapus data *goods* |
| **Aktor** | Super admin dan operator |
| **Kondisi Awal** | Sistem menampilkan data *goods* |
| **Kondisi Akhir** | Data berhasil hapus dari basisdata |
| **Skenario** | 1. *User* mengakses halaman data *goods*. 2. Sistem akan menampilkan data *goods.* 3. *User* memilih data yang akan dihapus 4. Sistem akan memberikan pesan konfirmasi untuk menghapus data tersebut. 5. *User* mengkofirmasi pesan untuk menghapus data teserbut. 6. Sistem akan memberikan pesan data telah berhasil di hapus. |
| **Skenario Alternatif** | Pada langka ke 4 user bisa menggagalkan data yang sudah dipilih untuk dirubah, dan sistem akan kembali ke langkah 2. |

Table 0‑9. Skenario Use Case Hapus Data Goods

* + 1. **Skenario Use Case Tambah Data Peminjaman**

|  |  |
| --- | --- |
| **Nama Use case** | **Mengolah data peminjaman** |
| **Deskripsi** | Super admin dan operator manmbahkan data peminjaman |
| **Aktor** | Super admin dan operator |
| **Kondisi Awal** | Sistem menampilkan *form* datapeminjaman baru |
| **Kondisi Akhir** | Data berhasil tambahkan dan tersimpan di basisdata |
| **Skenario** | 1. *User* mengakses *form* peminjaman baru. 2. Sistem akan menampilkan *form* peminjaman baru. 3. *User* memasukkan data sesuai dengan *form* yang ditampilkan sistem 4. Sistem menyimpan data *user* |
| **Skenario Alternatif** | Apabila ada *form* yang masih kosong *user* akan kembali ke langkah 3. |

Table 0‑10. Skenario Use Case Tambah Data Peminjaman

* + 1. **Skenario Use Case Ubah Data Peminjaman**

|  |  |
| --- | --- |
| **Nama Use case** | **Mengolah data peminjaman** |
| **Deskripsi** | Super admin dan operator mengubah data peminjaman |
| **Aktor** | Super admin dan operator |
| **Kondisi Awal** | Sistem menampilkan data peminjaman |
| **Kondisi Akhir** | Data berhasil diubah dan tersimpan di basisdata |
| **Skenario** | 1. *User* mengakses halaman data peminjaman. 2. Sistem akan menampilkan data peminjaman*.* 3. *User* memilih data yang akan diubah 4. Sistem akan menampilkan data sesuai dengan pilihan. 5. *User* merubah data yang sudah dipilih 6. *User* menyimpan data yang telah di rubah 7. Sistem akan memberikan pesan konfirmasi data berhasil di rubah |
| **Skenario Alternatif** | Pada langka ke 4 user bisa menggagalkan data yang sudah dipilih untuk dirubah, dan sistem akan kembali ke langkah 2. |

Table 0‑11. Skenario Use Case Ubah Data Peminjaman

* + 1. **Skenario Use Case Hapus Data Peminjaman**

|  |  |
| --- | --- |
| **Nama Use case** | **Mengolah data Peminjaman** |
| **Deskripsi** | Super admin dan operator menghapus datapeminjaman |
| **Aktor** | Super admin dan operator |
| **Kondisi Awal** | Sistem menampilkan data peminjaman |
| **Kondisi Akhir** | Data berhasil hapus dari basisdata |
| **Skenario** | 1. *User* mengakses halaman data peminjaman. 2. Sistem akan menampilkan data peminjaman. 3. *User* memilih data yang akan dihapus. 4. Sistem akan memberikan pesan konfirmasi untuk menghapus data tersebut. 5. *User* mengkofirmasi pesan untuk menghapus data teserbut. 6. Sistem akan memberikan pesan data telah berhasil di hapus. |
| **Skenario Alternatif** | Pada langka ke 4 user bisa menggagalkan data yang sudah dipilih untuk dirubah, dan sistem akan kembali ke langkah 2. |

Table 0‑12. Skenario Use Case Ubah Data Peminjaman

* + 1. **Skenario Use Case Tambah Data Pengembalian**

|  |  |
| --- | --- |
| **Nama Use case** | **Mengolah data pengembalian** |
| **Deskripsi** | Super admin dan operator manmbahkan data pengembalian |
| **Aktor** | Super admin dan operator |
| **Kondisi Awal** | Sistem menampilkan *form* datapengmbalian |
| **Kondisi Akhir** | Data berhasil tambahkan dan tersimpan di basisdata |
| **Skenario** | 1. *User* mengakses *form* peminjaman baru. 2. Sistem akan menampilkan *form* peminjaman baru. 3. *User* memasukkan data sesuai dengan *form* yang ditampilkan sistem 4. Sistem menyimpan data *user* |
| **Skenario Alternatif** | Apabila ada *form* yang masih kosong *user* akan kembali ke langkah 3. |

Table 0‑13. Skenario Use Case Tambah Data Pengembalian

* + 1. **Skenario Use Case Ubah Data Pengembalian**

|  |  |
| --- | --- |
| **Nama Use case** | **Mengolah data pengembalian** |
| **Deskripsi** | Super admin dan operator mengubah data pengembalian |
| **Aktor** | Super admin dan operator |
| **Kondisi Awal** | Sistem menampilkan data pengembalian |
| **Kondisi Akhir** | Data berhasil diubah dan tersimpan di basisdata |
| **Skenario** | 1. *User* mengakses halaman data pengembalian. 2. Sistem akan menampilkan data pengembalian*.* 3. *User* memilih data yang akan diubah 4. Sistem akan menampilkan data sesuai dengan pilihan. 5. *User* merubah data yang sudah dipilih 6. *User* menyimpan data yang telah di rubah 7. Sistem akan memberikan pesan konfirmasi data berhasil di rubah |
| **Skenario Alternatif** | Pada langka ke 4 user bisa menggagalkan data yang sudah dipilih untuk dirubah, dan sistem akan kembali ke langkah 2. |

Table 0‑14. Skenario Use Case Ubah Data Pengembalian

* + 1. **Skenario Use Case Menghapus Data Pengembalian**

|  |  |
| --- | --- |
| **Nama Use case** | **Mengolah data Pengembalian** |
| **Deskripsi** | Super admin dan operator menghapus datapengembalian |
| **Aktor** | Super admin dan operator |
| **Kondisi Awal** | Sistem menampilkan data pengembalian |
| **Kondisi Akhir** | Data berhasil hapus dari basisdata |
| **Skenario** | 1. *User* mengakses halaman data pengembalian. 2. Sistem akan menampilkan data pengembalian. 3. *User* memilih data yang akan dihapus. 4. Sistem akan memberikan pesan konfirmasi untuk menghapus data tersebut. 5. *User* mengkofirmasi pesan untuk menghapus data teserbut. 6. Sistem akan memberikan pesan data telah berhasil di hapus. |
| **Skenario Alternatif** | Pada langka ke 4 user bisa menggagalkan data yang sudah dipilih untuk dirubah, dan sistem akan kembali ke langkah 2. |

Table 0‑15. Skenario Use Case Menghapus Data Pengembalian

* + 1. **Skenario Use Case *Eskpor* Laporan**

|  |  |
| --- | --- |
| **Nama Use case** | ***Eskpor* Laporan** |
| **Deskripsi** | Super admin dan operator manmbahkan data pengembalian |
| **Aktor** | Super admin dan operator |
| **Kondisi Awal** | Sistem menampilkan data laporan |
| **Kondisi Akhir** | Laporan berhasil di *eskpor* |
| **Skenario** | 1. *User* mengakses halaman laporan. 2. Sistem akan menampilkan laporan. 3. *User* memilih menu *eskpor* laporan 4. Sistem mencetak laporan |
| **Skenario Alternatif** | - |

Table 0‑16. Skenario Use Case Eskpor Laporan

* + 1. **Skenario Use Case Tambah data Kalibrasi *Goods***

|  |  |
| --- | --- |
| **Nama Use case** | **Kalibrasi *Goods*** |
| **Deskripsi** | Super admin dan kalibrator manmbahkan data kalibrasi |
| **Aktor** | Super admin dan kalibrator |
| **Kondisi Awal** | Sistem menampilkan *form* datakalibrasi |
| **Kondisi Akhir** | Data berhasil tambahkan dan tersimpan di basisdata |
| **Skenario** | 1. *User* mengakses *form* kalibrasi. 2. Sistem akan menampilkan *form* kalibrasi. 3. *User* memasukkan data sesuai dengan *form* yang ditampilkan sistem 4. Sistem menyimpan data *user* |
| **Skenario Alternatif** | - |

Table 0‑17. Skenario Use Case Tambah data Kalibrasi Goods

* + 1. **Skenario Use Case *Eskpor* Laporan Kalibarasi**

|  |  |
| --- | --- |
| **Nama Use case** | ***Eskpor* Laporan** |
| **Deskripsi** | Super admin, operator dan kalibrator manmbahkan data pengembalian |
| **Aktor** | Super admin, operator dan kalibrator |
| **Kondisi Awal** | Sistem menampilkan data laporan |
| **Kondisi Akhir** | Laporan kalibrasi berhasil di *eskpor* |
| **Skenario** | 1. *User* mengakses halaman laporan *da*. 2. Sistem akan menampilka laporan. 3. *User* memilih menu *eskpor* laporan 4. Sistem mencetak laporan |
| **Skenario Alternatif** | - |

Table 0‑18. Skenario Use Case Eskpor Laporan Kalibarasi

* 1. **Sequence Diagram**

1. **Sequence Diagram Login**

Gambar 2. Sequence Diagram Login

1. **Sequence Diagram Tambah Data *User***

Gambar 3. Sequence Diagram Tambah Data User

1. **Sequence Diagram Ubah Data *User***

Gambar 4. Sequence Diagram Ubah Data *User*

1. **Sequence Diagram Hapus *User***

Gambar 5. Sequence Diagram Hapus User

1. **Sequence Diagram Tambah *Goods***

Gambar 6. Sequence Diagram Tambah Goods

1. **Sequence Diagram Ubah *Goods***

Gambar 7. Sequence Diagram Ubah Goods

1. **Sequence Diagram Hapus *Goods***

Gambar 8. Sequence Diagram Hapus Goods

1. **Sequence Diagram Tambah Peminjaman**

Gambar 9. Sequence Diagram Tambah Peminjaman

1. **Sequence Diagram Ubah Peminjaman**

Gambar 10. Sequence Diagram Tambah Peminjaman

1. **Sequence Diagram Hapus Peminjaman**



Gambar 11. Sequence Diagram Hapus Peminjaman

1. **Sequence Diagram Tambah Pengembalian**



Gambar 12. Sequence Diagram Tambah Pengembalian

1. **Sequence Diagram Ubah Pengembalian**



Gambar 13. Sequence Diagram Ubah Pengembalian

1. **Sequence Diagram Hapus Pengembalian**



Gambar 14. Sequence Diagram Hapus Pengembalian

1. **Sequence Diagram Eskpor Data**



Gambar 15. Sequence Diagram Eskpor Data

1. **Sequence Diagram Tambah Data Kalibrasi**

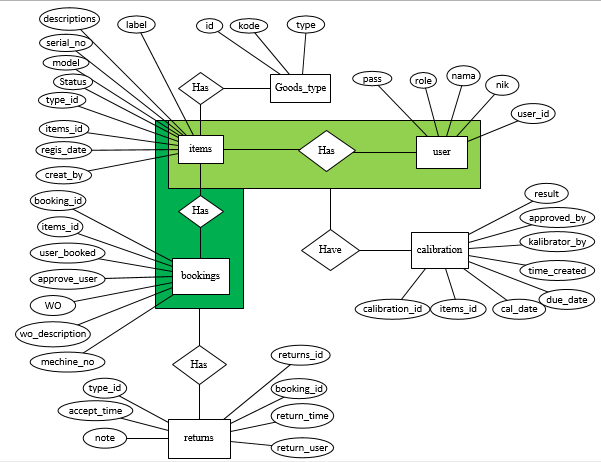


Gambar 16. Sequence Diagram Tambah Data Kalibrasi

1. **Sequence Diagram Eskpor Data Kalibrasi**



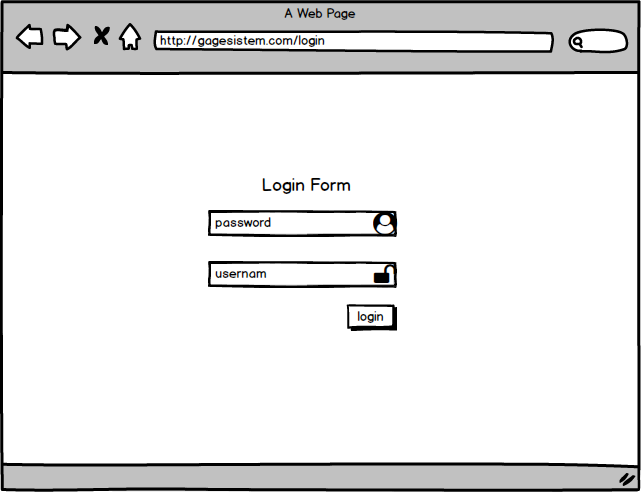
Gambar 17. Sequence Diagram Eskpor Data Kalibrasi

* 1. **ER Diagram**

Gambar 18. ER Diagram

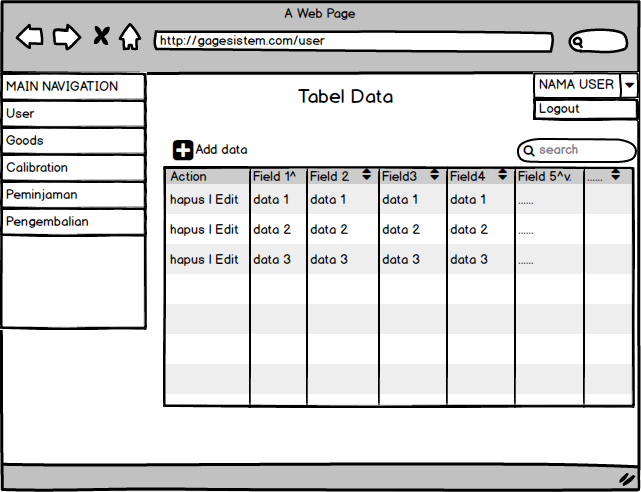
* 1. **Desain *User Interface***

1. ***Tampilan Form login***

****

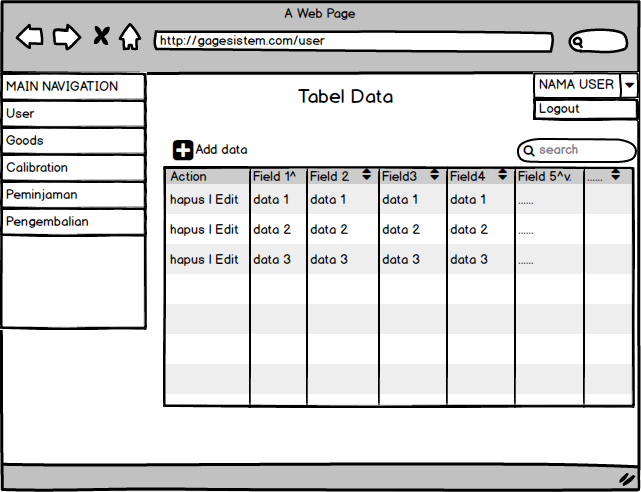
Gambar 19. Tampilan Form Login

1. ***User Interface* Tampil Data**

****

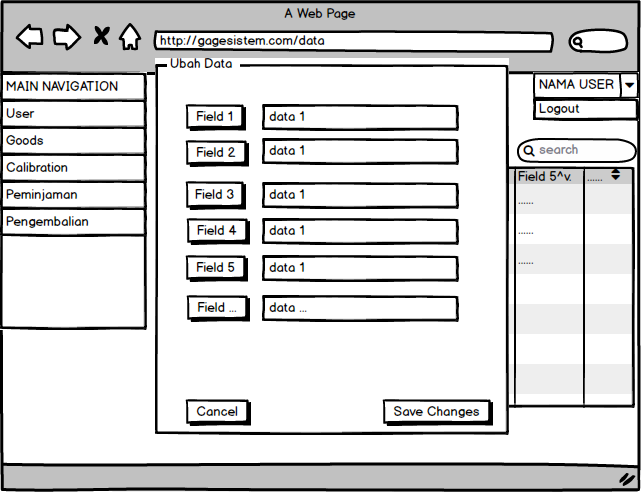
Gambar 20. User Interface Tampil Data

1. ***User Interface* Tambah Data**

****

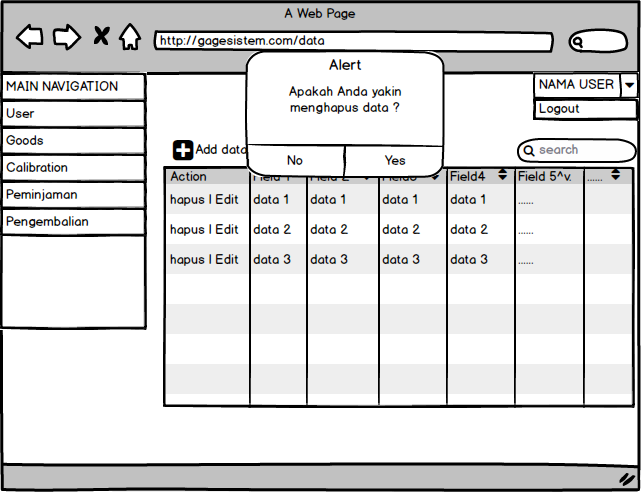
Gambar 21. User Interface Tambah Data

1. ***User Interface* Ubah Data**

****

Gambar 22. User Interface Ubah Data

1. ***User Interface* Hapu Data**

****

Gambar 23. User Interface Delete Data

