BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1. Implementasi

Implementasi merupakan tahap lanjutan dari kegiatan perancangan sistem, tahap ini merupakan tahap meletakan sistem supaya siap untuk di operasikan dan dapat di pandang sebagai usaha untuk mewujudkan sistem yang telah di rancang. Langkah-langkah dalam tahap implementasi ini adalah urutan kegiatan awal sampai akhir yang harus dilakukan dalam mewujudkan sistem yang telah dirancang.

5.1.1. Batasan Implementasi

Dalam rencana penerapan sistem yang terkomputerisasi yakni agar sistem siap di operasikan, maka perlu di adakan kegiatan-kegiatan dari penerapannya. Adapun langkah-langkah yang perlu dilakukan untuk menerapkan sistem tersebut adalah pembuatan program, testing program, pelatihan serta terima dokumentasi, akan tetapi dalam penulisan tugas akhir ini implementasi sistem yang dilakukan hanya sampai pada tahap pembuatan program dan pengujian program.

5.1.2. Implementasi Perangkat Lunak

Untuk mendukung kelancaran sistem informasi yang di rancang, maka sistem ini memerlukan perangkat lunak. Perangkat lunak digunakan untuk

mendukung kinerja *database.* Adapun pernagkat lunak yang digunakan adalah sebagai berikut :

1. Sistem Operasi Windows XP atau Windows 7
2. Program Aplikasi Web Browser
3. Program Aplikasi Database MySQL

5.1.3. Implementasi Perangkat Keras

Perangkat keras yang dibutuhkan berdasarkan kebutuhan minimal yang harus terpenuhi untuk dapat menjalankan sistem aplikasi ini yaitu:

1. Menggunakan minimal processor Intel Pentium 4
2. RAM Minimal 2 GB (jika memakai windows 7)
3. Hard Disk minimal 10GBuntuk media penyimpanan.
4. *Mouse, keyboard,* Monitor color 14 inc sebagai alat antar muka.
5. *Printer.*

5.1.4. Implementasi Basis Data (Sintaks SQL)

Implementasi basis data digunakan untuk membuat struktur database dan tabel yang akan digunakan untuk sistem informasi di Mr.Clean, berikut adalah sintak Sql yang digunakan :

Implementasi basis data digunakan untuk membuat struktur database dan tabel yang akan digunakan untuk sistem informasi di Mr.Clean, berikut adalah sintak Sql yang digunakan :

1. **Tabel Item**

```
CREATE TABLE [dbo].[Item](
      [Code] [varchar](10) NOT NULL,
      [Name] [varchar](50) NOT NULL,
      [OrderLevel] [float] NOT NULL,
      [Notes] [varchar](200) NULL,
      [IsDeleted] [bit] NOT NULL,
      [IsProductComponent] [bit] NOT NULL,
 CONSTRAINT [PK_Item] PRIMARY KEY CLUSTERED
(
      [Code] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS  = ON,
ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
```

2. **Table Product**

```
CREATE TABLE [dbo].[Product](
      [Code] [varchar](10) NOT NULL,
      [Name] [varchar](50) NOT NULL,
      [UnitPrice] [float] NOT NULL,
      [IsDeleted] [bit] NOT NULL,
 CONSTRAINT [PK_Product] PRIMARY KEY CLUSTERED
(
      [Code] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS  = ON,
ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
```

3. **Table Period**

```
CREATE TABLE [dbo].[Period](
      [Code] [varchar](6) NOT NULL,
      [IsOpen] [bit] NOT NULL,
 CONSTRAINT [PK_Period] PRIMARY KEY CLUSTERED
(
      [Code] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS  = ON,
ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
```

4. Tabel Notification

```
CREATE TABLE [dbo].[Notification](
	[NotifyDate] [datetime] NOT NULL,
	[Seq] [smallint] NOT NULL,
	[MailTo] [varchar](200) NOT NULL,
	[MailCC] [varchar](200) NULL,
	[Subject] [varchar](200) NULL,
	[Body] [text] NULL,
	[Sent] [bit] NOT NULL
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
```

5. Table Customer

```
CREATE TABLE [dbo].[Customer](
	[Code] [varchar](10) NOT NULL,
	[Name] [varchar](50) NOT NULL,
	[Address] [varchar](200) NULL,
	[Phone] [varchar](20) NULL,
	[Email] [varchar](50) NULL,
	[Notes] [varchar](500) NULL,
	[IsDeleted] [bit] NOT NULL,
 CONSTRAINT [PK_Customer] PRIMARY KEY CLUSTERED
(
	[Code] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS  = ON,
ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
```

6. Table Purchase

```
CREATE TABLE [dbo].[Purchase](
	[Code] [varchar](20) NOT NULL,
	[PurchaseDate] [datetime] NOT NULL,
	[Seller] [nvarchar](50) NULL,
	[Amount] [money] NOT NULL,
	[IsDeleted] [bit] NOT NULL,
 CONSTRAINT [PK_Purchase] PRIMARY KEY CLUSTERED
(
	[Code] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS  = ON,
ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
```

7. Table REF_LoginType

```
CREATE TABLE [dbo].[REF_LoginType](
	[Id] [smallint] NOT NULL,
	[Name] [varchar](20) NOT NULL,
 CONSTRAINT [PK_REF_LoginType] PRIMARY KEY CLUSTERED
(
	[Id] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS  = ON,
ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
```

8. Table REF_Company

```
CREATE TABLE [dbo].[REF_Company](
	[CompanyName] [varchar](50) NOT NULL,
	[Address] [varchar](200) NULL,
	[Phone] [varchar](20) NULL,
	[Email] [varchar](50) NULL,
 CONSTRAINT [PK_REF_Company] PRIMARY KEY CLUSTERED
(
	[CompanyName] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS  = ON,
ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
```

9. Table Purchase Detail

```
CREATE TABLE [dbo].[PurchaseDetail](
	[PurchaseCode] [varchar](20) NOT NULL,
	[Seq] [smallint] NOT NULL,
	[ItemCode] [varchar](10) NOT NULL,
	[Qty] [float] NOT NULL,
	[UnitPrice] [money] NOT NULL,
	[IsDeleted] [bit] NOT NULL,
 CONSTRAINT [PK_PurchaseDetail] PRIMARY KEY CLUSTERED
(
	[PurchaseCode] ASC,
	[Seq] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS  = ON,
ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
```

10. Table Product Cost

```
CREATE TABLE [dbo].[ProductCost](
	[ProductCode] [varchar](10) NOT NULL,
	[ItemCode] [varchar](10) NOT NULL,
	[Qty] [float] NOT NULL,
	[UnitPrice] [money] NOT NULL,
 CONSTRAINT [PK_ProductCost] PRIMARY KEY CLUSTERED
(
	[ProductCode] ASC,
	[ItemCode] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS  = ON,
ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
```

11. Table Item Balance

```
CREATE TABLE [dbo].[ItemBalance](
	[ItemCode] [varchar](10) NOT NULL,
	[PeriodCode] [varchar](6) NOT NULL,
	[QtyIn] [float] NOT NULL,
	[QtyOut] [float] NOT NULL,
	[QtyAdj] [float] NOT NULL,
 CONSTRAINT [PK_ItemBalance_1] PRIMARY KEY CLUSTERED
(
	[ItemCode] ASC,
	[PeriodCode] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS  = ON,
ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
```

12. Table Employee

```
CREATE TABLE [dbo].[Employee](
	[Code] [varchar](10) NOT NULL,
	[Name] [varchar](50) NOT NULL,
	[Address] [varchar](200) NULL,
	[Phone] [varchar](20) NULL,
	[Email] [varchar](50) NULL,
	[Notes] [varchar](200) NULL,
	[IsAllowLogin] [bit] NOT NULL,
	[LoginTypeId] [smallint] NULL,
	[Username] [varchar](20) NULL,
	[Password] [nvarchar](8) NULL,
	[IsDeleted] [bit] NOT NULL,
 CONSTRAINT [PK_Employee] PRIMARY KEY CLUSTERED
(
	[Code] ASC
```

```
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS  = ON,
ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
```

13. Table Sales

```
CREATE TABLE [dbo].[Sales](
      [Code] [varchar](10) NOT NULL,
      [SalesDate] [datetime] NOT NULL,
      [EmployeeCode] [varchar](10) NULL,
      [CustomerCode] [varchar](10) NULL,
      [CustomerName] [varchar](50) NULL,
      [Address] [varchar](200) NULL,
      [Phone] [varchar](20) NULL,
      [ItemsCount] [int] NOT NULL,
      [Amount] [money] NOT NULL,
      [PaidAmount] [money] NOT NULL,
      [IsDeleted] [bit] NOT NULL,
 CONSTRAINT [PK_Sales_1] PRIMARY KEY CLUSTERED
(
      [Code] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS  = ON,
ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
```

14. Table Sales Detail

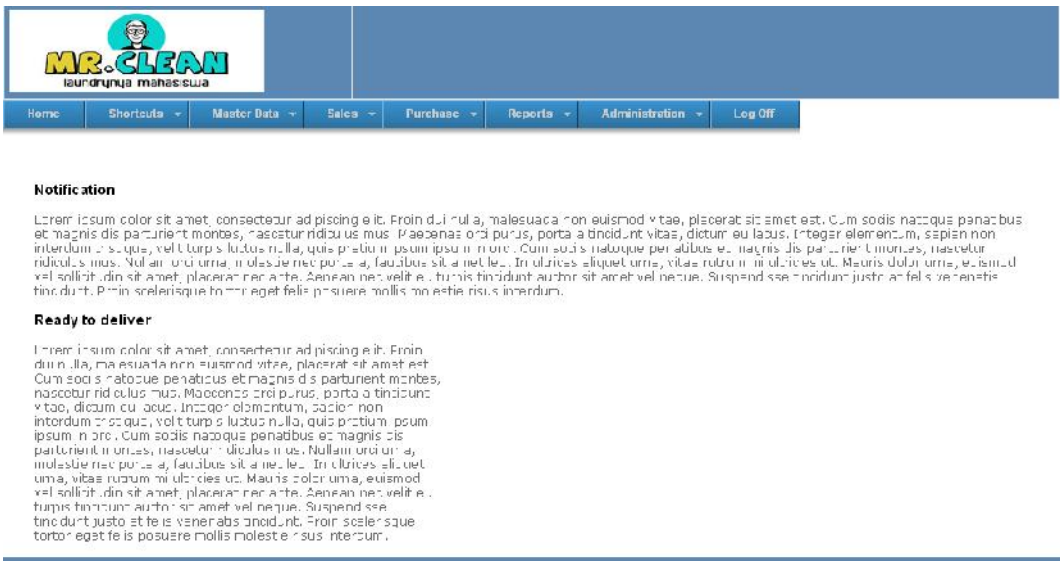
```
CREATE TABLE [dbo].[SalesDetail](
      [SalesCode] [varchar](10) NOT NULL,
      [ProductCode] [varchar](10) NOT NULL,
      [Seq] [smallint] NOT NULL,
      [Qty] [float] NOT NULL,
      [UnitPrice] [money] NOT NULL,
      [IsDeleted] [bit] NOT NULL,
      [RowId] [bigint] IDENTITY(1,1) NOT NULL,
 CONSTRAINT [PK_SalesDetail] PRIMARY KEY CLUSTERED
(
      [SalesCode] ASC,
      [ProductCode] ASC
)WITH (PAD_INDEX  = OFF, STATISTICS_NORECOMPUTE  = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS  = ON,
ALLOW_PAGE_LOCKS  = ON) ON [PRIMARY]
) ON [PRIMARY]
```

5.1.5. Implementasi Antar Muka

Pada tahapan ini akan diterangkan secara singkat penggunaan program sistem informasi di Mr.Clean beserta cara penggunaanya sebagai berikut:

1. Menu Utama

   Form menu utama Mr.Clean merupakan tampilan yang menampung form-form informasi. Berikut adalah tampilan form yang dimaksud.



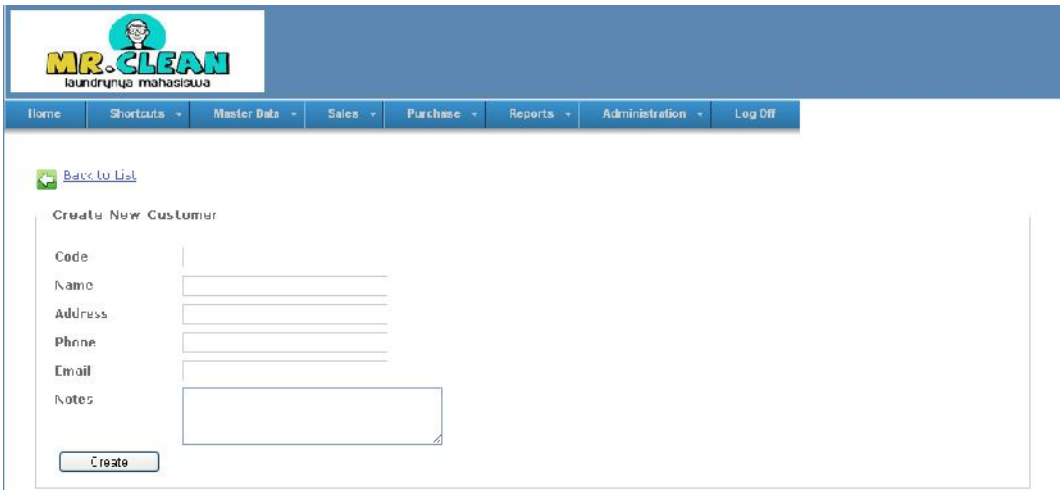
Gambar 5.1. Form menu utama Mr.Clean

2. Halaman *Customer*



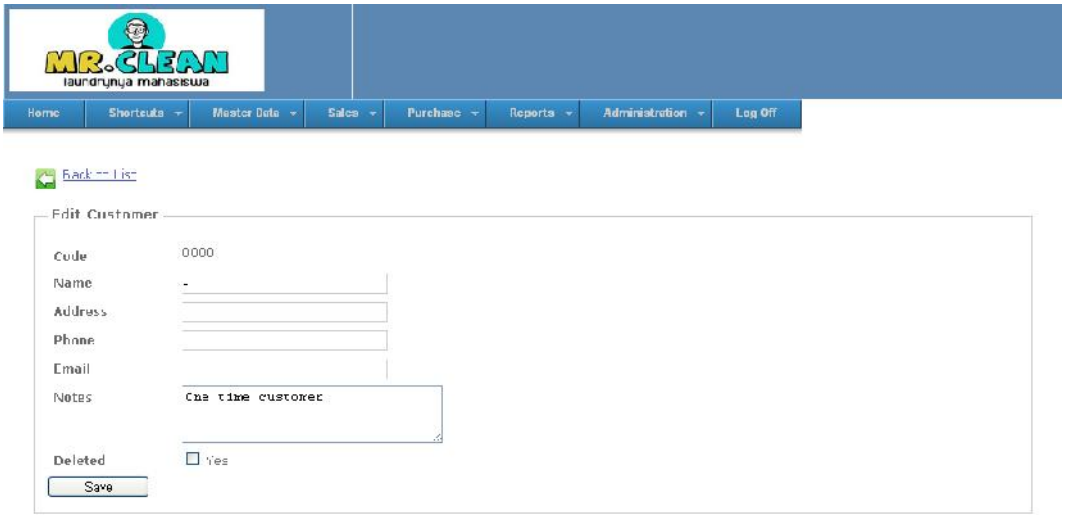
Gambar 5.2. Form halaman *customer*

3. Halaman *Create New Customer*



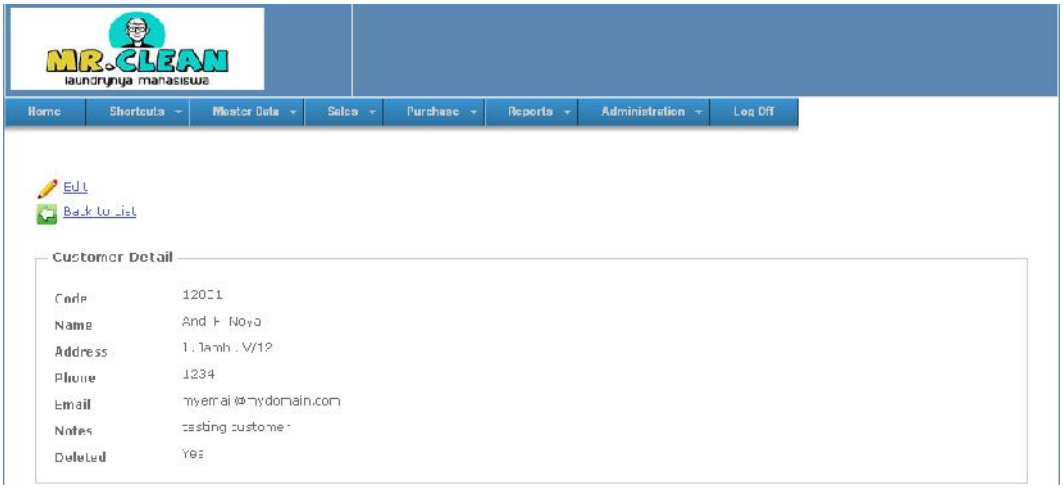
Gambar 5.3. Form *Create New Customer*

4. Halaman *Edit Customer*



Gambar 5.4. Form *Edit Customer*

5. Halaman *Customer Detail*



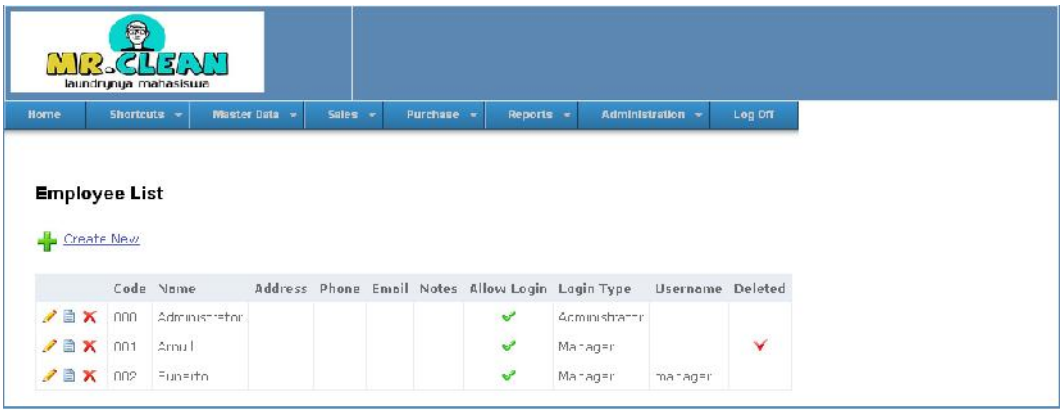
Gambar 5.5. Form *Customer Detail*

6. Halaman *Deleting Customer*



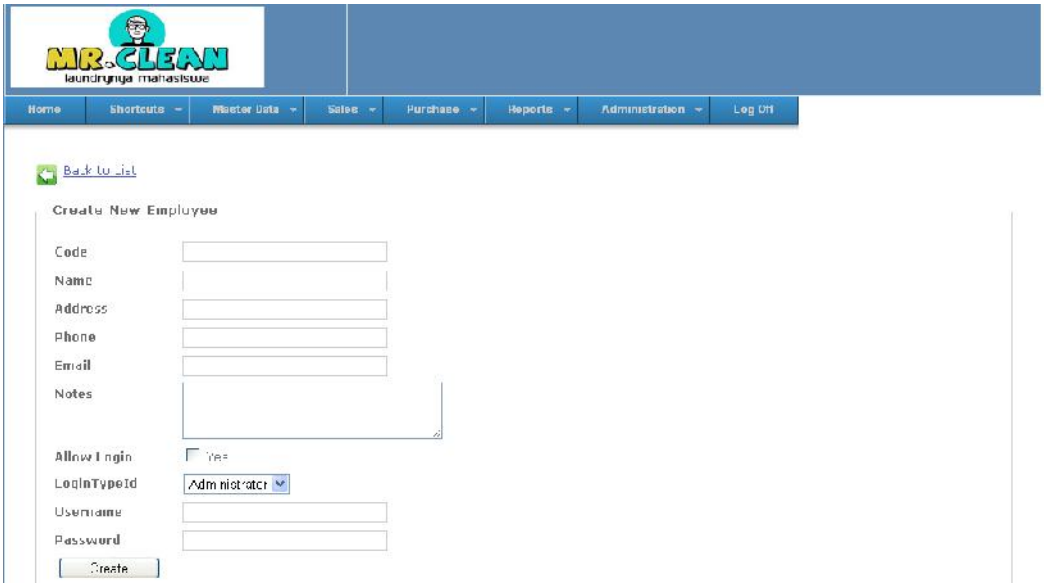
Gambar 5.6. Halaman *Deleting Customer*

7. Halaman *Employee*



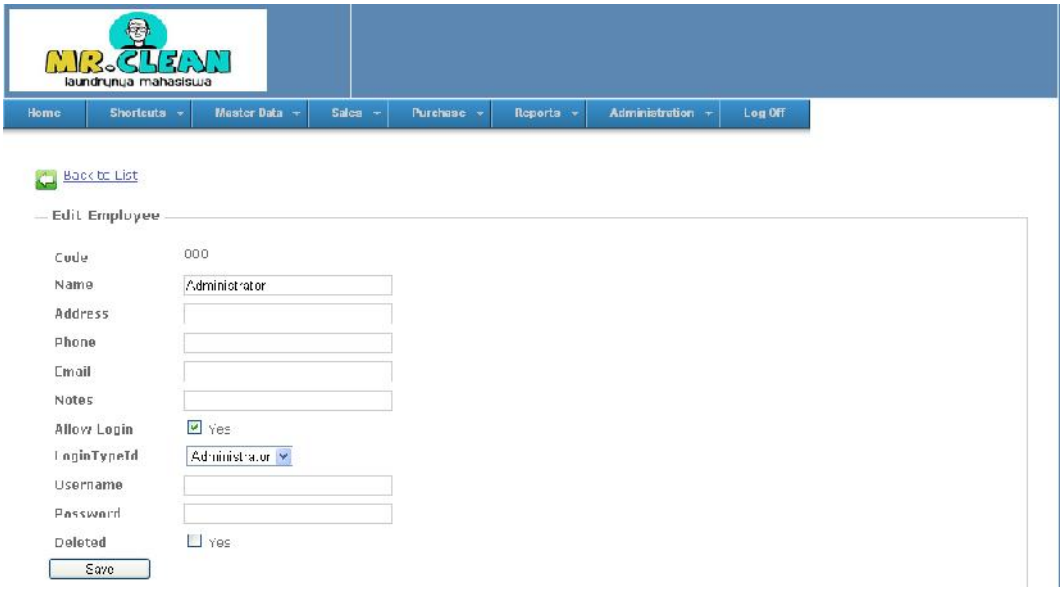
Gambar 5.7. Halaman *Employee*

8. Halaman *Create New Employee*



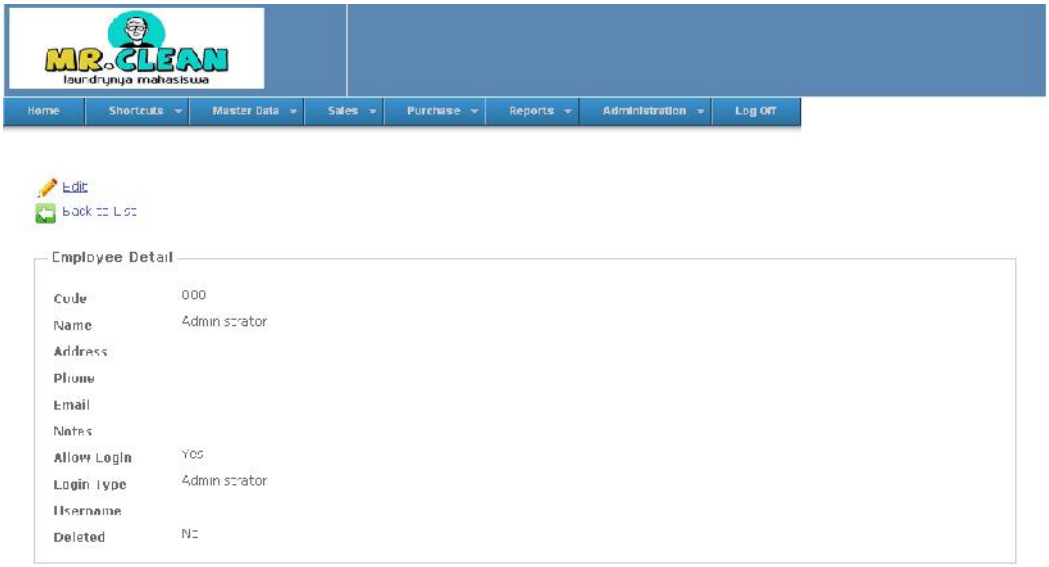
Gambar 5.8. Halaman *Create New Employee*

9. Halaman *Edit Employee*



Gambar 5.9. Halaman *Edit Employee*

10. Halaman *Employee Detail*



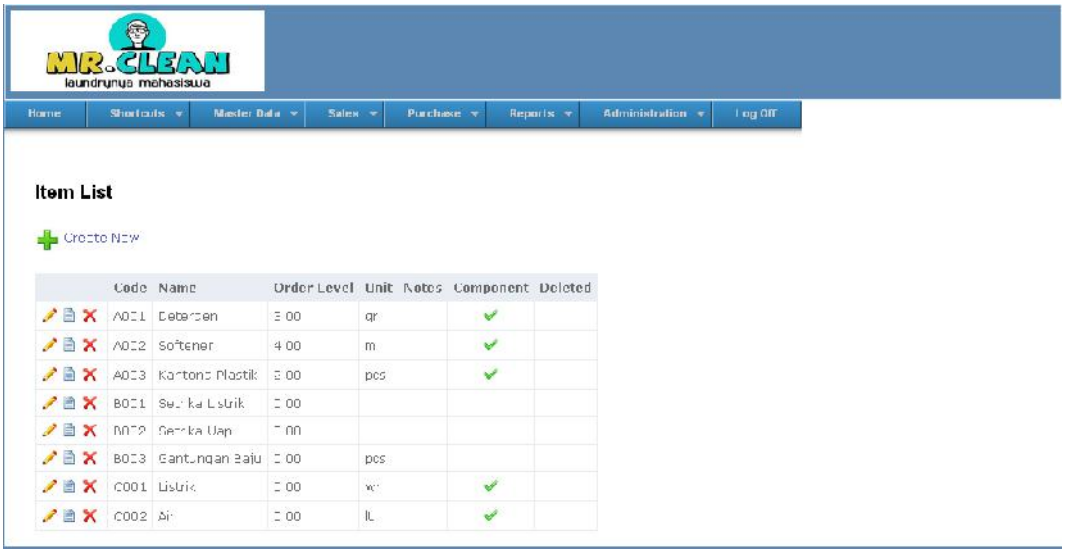
Gambar 5.10. Halaman *Employee Detail*

11. Halaman *Deleting Employee*



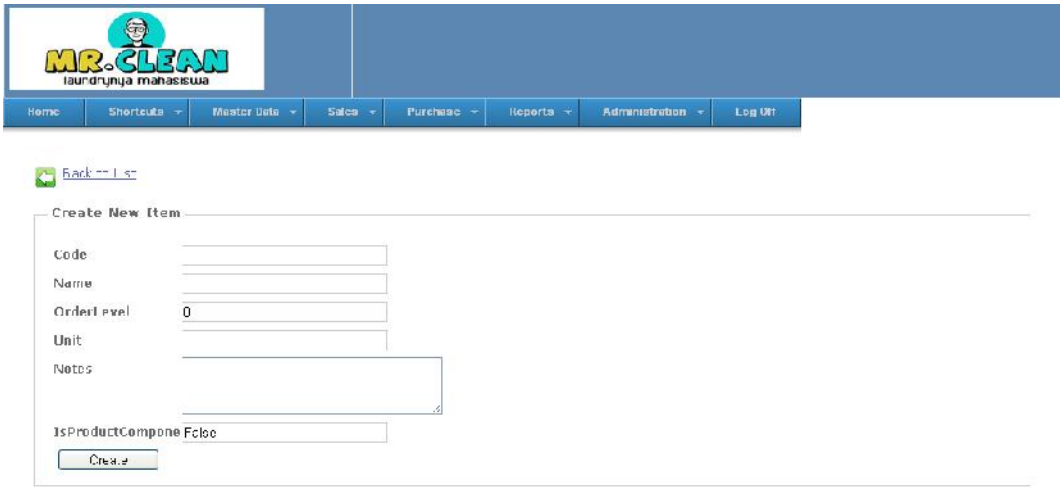
Gambar 5.11. Halaman *Deleting Employee*

12. Halaman *Item*



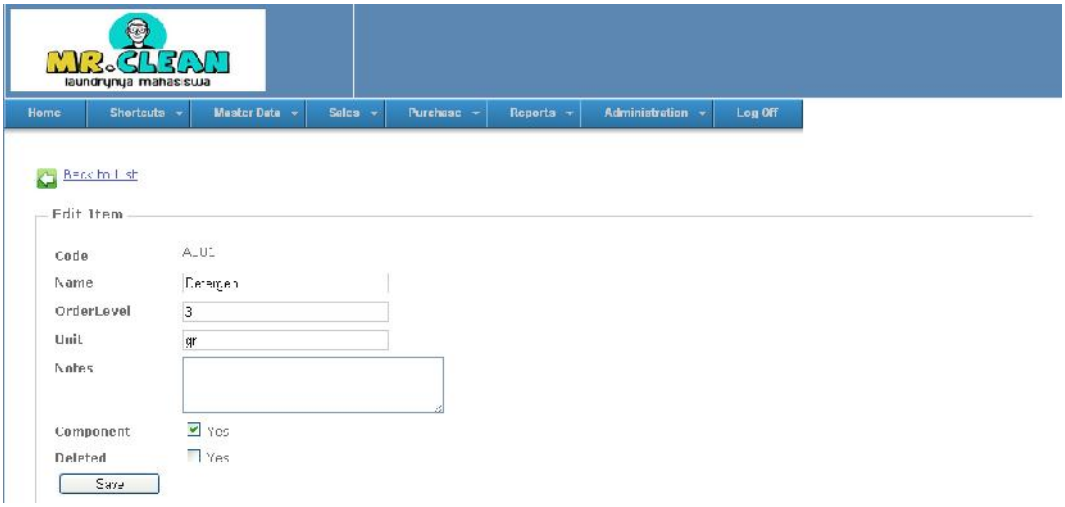
Gambar 5.12. Halaman *Item*

13. Halaman *Create New Item*



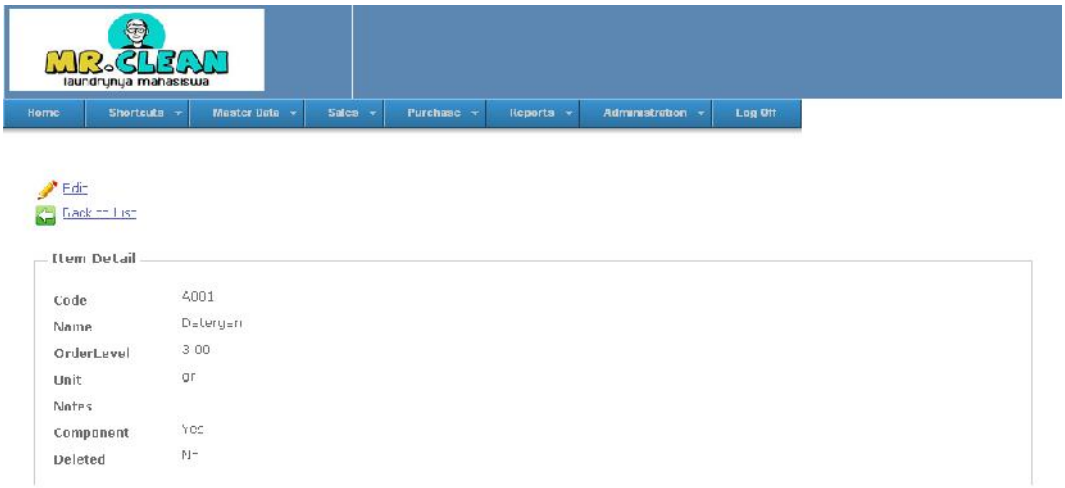
Gambar 5.13. Halaman *Create New Item*

14. Halaman *Edit Item*



Gambar 5.14. Halaman *Edit Item*

15. Halaman *Item Detail*



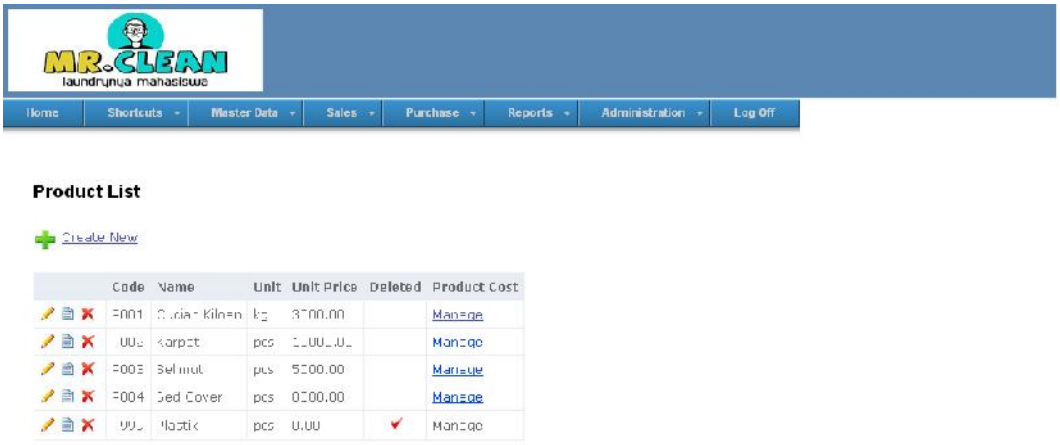
Gambar 5.15. Halaman *Item Detail*

16. Halaman *Deleting Item*



Gambar 5.16. Halaman *Deleting Item*

17. Halaman *Product*



Gambar 5.17. Halaman *Product*

18. Halaman *Create New Product*



Gambar 5.18. Halaman *Create New Product*

19. Halaman *Edit Product*



Gambar 5.19. Halaman *Edit Product*

20. Halaman *Product Detail*



Gambar 5.20. Halaman *Product Detail*

21. Halaman *Deleting Product*



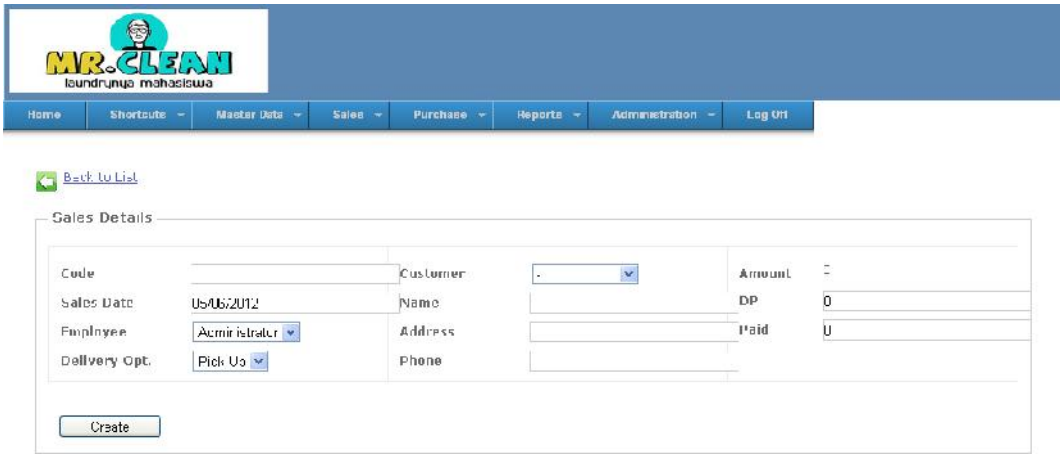
Gambar 5.21. Halaman *Deleting Product*

22. Halaman *Sales*



Gambar 5.22. Halaman *Sales*

23. Halaman *Create New Sales*



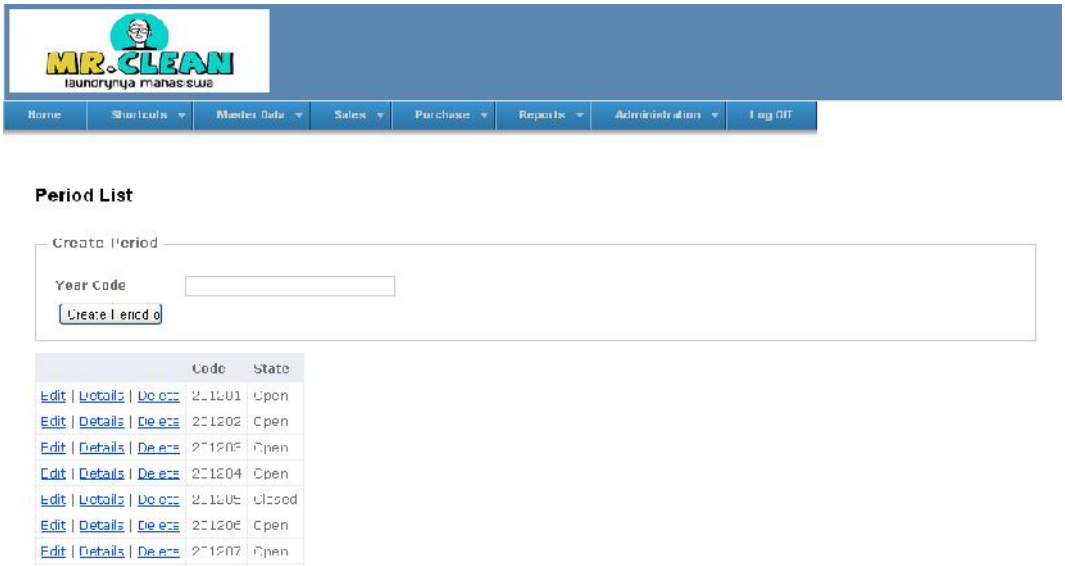
Gambar 5.23. Halaman *Create New Sales*

24. Halaman *Sales Detail*



Gambar 5.24. Halaman *Sales Detail*

25. Halaman *Period*



Gambar 5.25. Halaman *Period*

26. Halaman *Company*



Gambar 5.26. Halaman *Company*

5.1.6. ImplementasiInstalasi Program

Langkah langkah instalasi program / perangkat lunak:

1. Login kedalam *hosting* menggunakan *web browser* dengan mengisikan *username* dan *password* yang telah diberikan oleh penyedia *hosting* saat mendaftar kemudian tekan tombol*submit*. Penulis menggunakan *hosting* yang gratis saat implementasi perangkat lunak, hosting yang digunakan http://www.000webhost.com/

Gambar 5.27 Halaman *login* kedalam *cPanel hosting*



2. Setelah proses login maka akan tampil halaman utama pada *cPanel hosting* yang berisikan seluruh konfigurasi untuk perangkat lunak yang telah dibangun



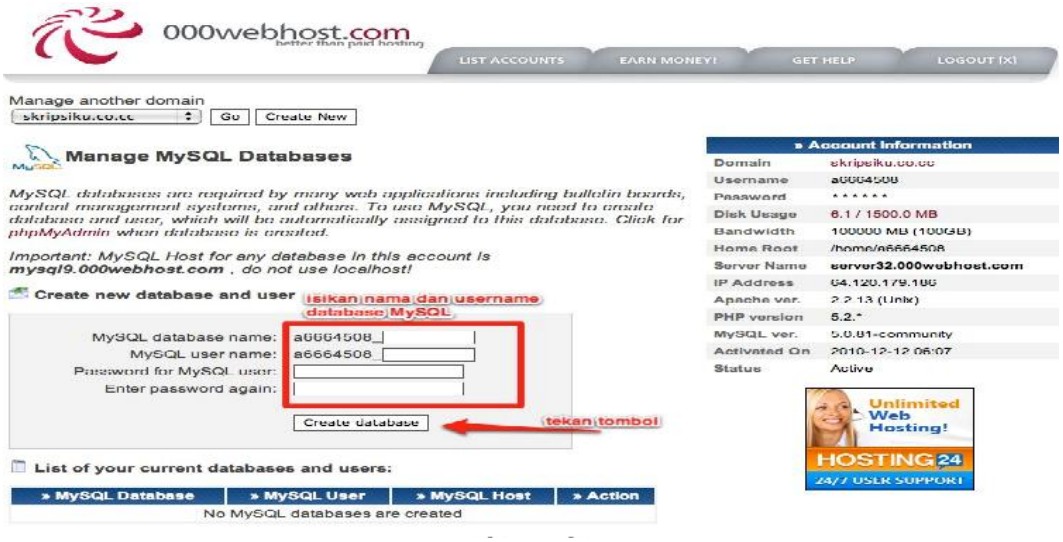
Gambar 5.28 Halaman utama cPanel

3. Setelah tampil halaman utama pada cPanel kemudian langkah berikutnya yang harus dilakukan yaitu membuat *database* pada *hosting* dengan memilih / menekan MySQL pada *cPanel* yg ada di *hosting*



Gambar 5.29 MySQL pada *hosting*

4. Setelah masuk ke halaman *MySQL* langkah berikutnya yaitu dengan memasukkan nama, *username*, dan *password* pada database *MySQL* kemudian tekan tombol *create database*



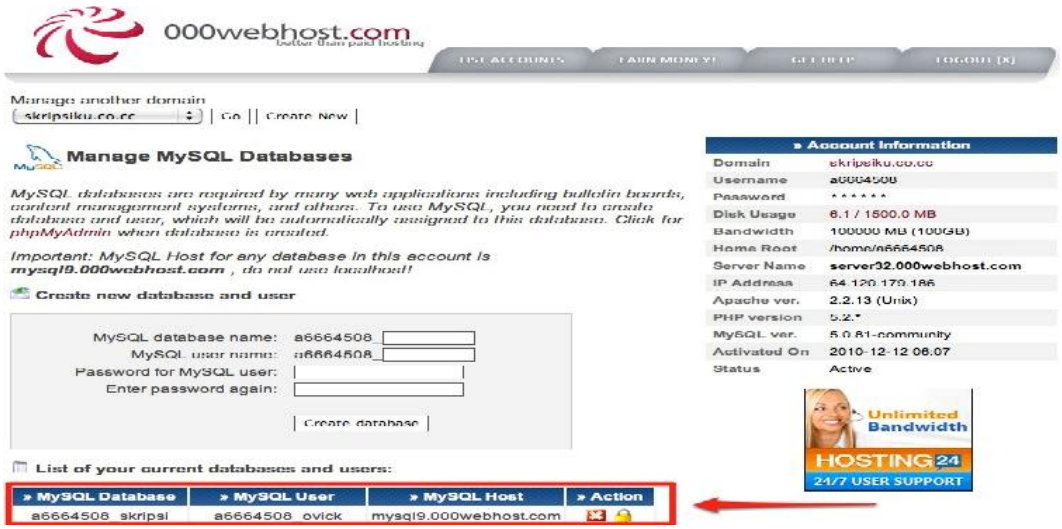
Gambar 5.30 Input nama, *username*, dan *passworddatabase MySQL*

5. Setelah memasukan nama, *username*, dan *password MySQL* selesai makan akan tampil pemberitahuan bahwa nama, *username*, dan *password* selesai dibuat kemudian simpanlah konfigurasi yang telah di inputkan tadi sebagai penyesuaian dengan perangkat lunak.



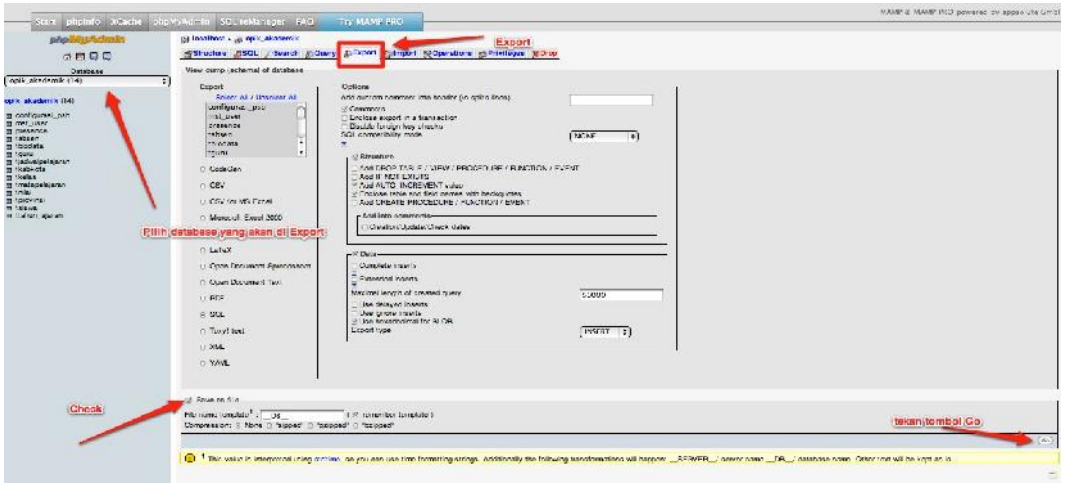
Gambar 5.31 Informasi konfigurasi nama, *username*, dan *password database*

6. Setelah proses input nama, *username*, dan *passwordMySQL*maka untuk memastikan bahwa sudah masuk kedalam *database* maka akan terlihat tabel seperti dibawah ini



Gambar 5.32 Tabel informasi konfigurasi nama, *username*, dan *hostMySQL*

7. Setelah proses diatas maka langkah berikutnya yang harus dilakukan yaitu dengan memindahkan *database* yang ada di *server* lokal kedalam *server* yang berada didalam *hosting* dengan meng*exportdatabase* yang berisikan *field-field*.



Gambar 5.33*Exportdatabaseserver* lokal

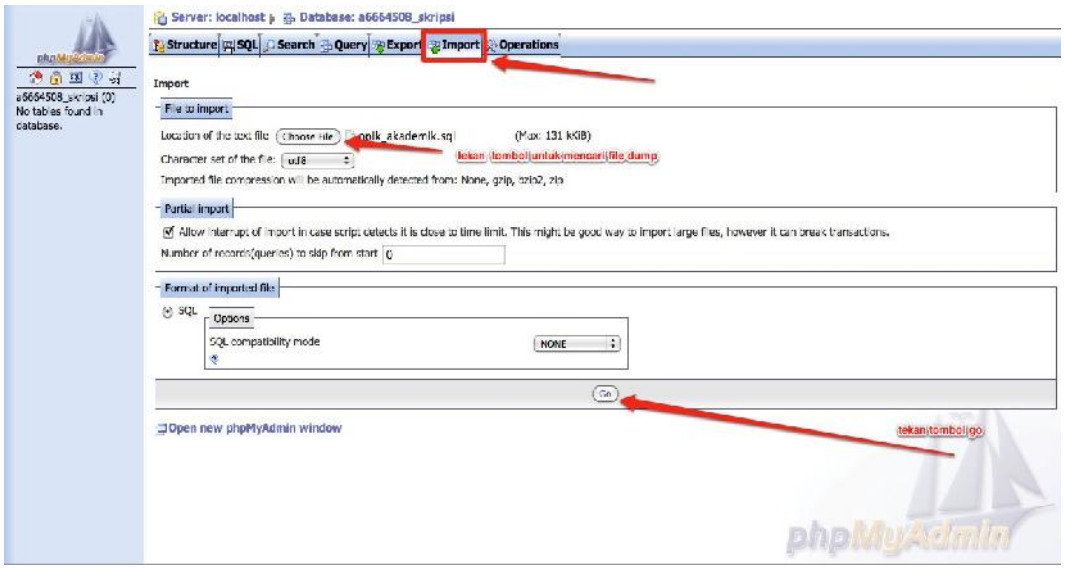
8. Setelah proses *export database* maka akan menghasilkan *dump* dan file tersebut berekstensi .sql

9. Setelah proses *export* maka masuk masuk ke halaman *phpMyAdmin* yang ada didalam *hosting*



Gambar 5.34*Php MyAdmin* didalam *hosting*

10. Setelah proses *export* dari *server* lokal langkah selanjutnya dengan meng*import*kan *file* sql dump yang berekstensi .sql kedalam *php MyAdmin* yang ada di *hosting*



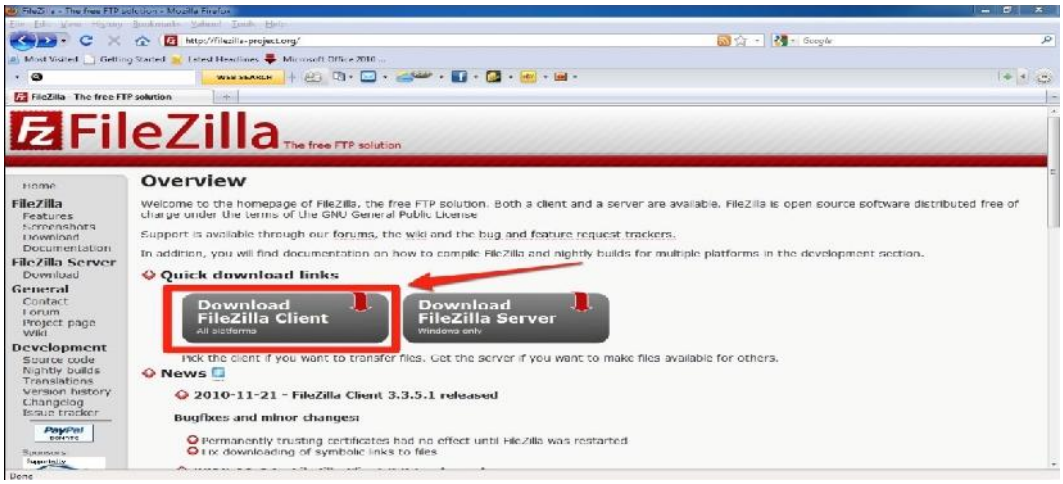
Gambar 5.35*Export sql dump* kedalam *hosting*

11. Langkah setelah proses *import database* dari *server* lokal kedalam *serverhosting* kemudian melakukan konfigurasi yang ada didalam salah satu file perangkat lunak
12. *Setting* kembali *file* konfigurasi akademik_ddy/config/koneksi.php perangkat lunak yang ada didalam *server* lokal sesuai nama, *username*,dan *password* yang telah dibuat tadi didalam *database hosting* pada langkah 5



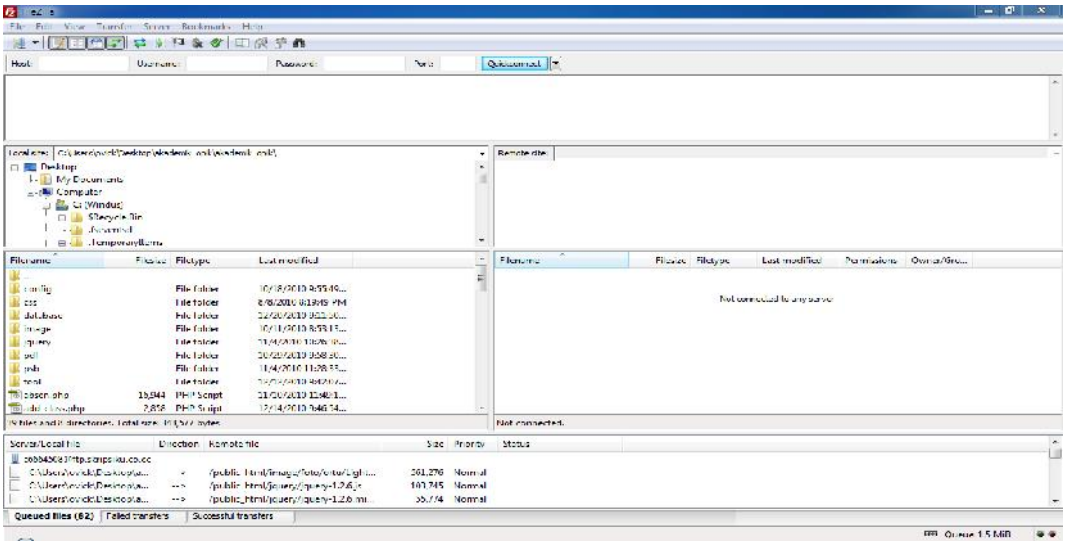
Gambar 5.36*File* konfigurasi

13. Unduh aplikasi *FTP* (*file transfer protocol*) *Client* bernama *FileZilla* di http://filezilla-project.org/, aplikasi ini sebagai pengunggah seluruh *file* program kedalam *hosting*.



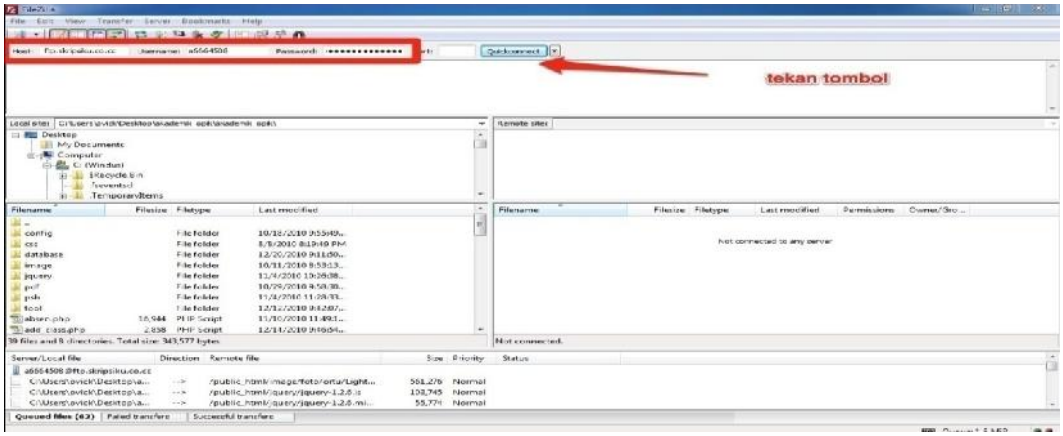
Gambar 5.37 Unduh Aplikasi *FileZilla*

14. Install aplikasi *FileZilla* kedalam sistem
15. Setelah proses *install* maka bukalah aplikasi tersebut, tampilan pertama akan terlihat seperti gambar dibawah ini



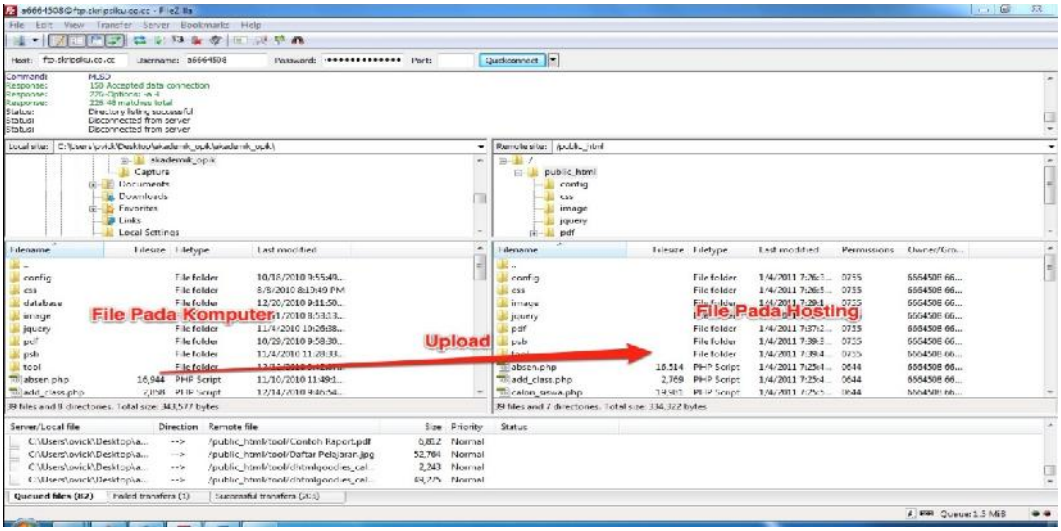
Gambar 5.38 Tampilan Utama *FileZilla*

16. Lakukan proses login kedalam hosting dengan aplikasi *FileZilla* kemudian masukkan *Host* dengan alamat *web*, *Username*, dan *password* sesuai inormasi yang telah diberikan oleh *hosting*, kemudian tekan tombol *Quickconnect*.



Gambar 5.39 Proses *login hosting*

17. Setelah *login* kedalam *hosting* kemudian *upload* seluruh *file* dan *folder* perangkat lunak sistem informasi akademik yang ada didalam komputer.



Gambar 5.40 Proses *upload file* kedalam *hosting*

18. Setelah *upload file* selesai maka untuk membuktikan apakah dapat dijalankan dan dapat melakukan koneksi ke *database*apa tidaknya perangkat lunak tersebut harus di test secara *detail*.
19. Untuk membuktikan apakah perangkat lunak dapat dijalankan di internet maka ketikkan di addres bar http://www.namadomain/.

5.2. Pengujian

Pengujian merupakan bagian yang penting dalam siklus pembangunan perangkat lunak. Pengujian di lakukan untuk menjamin kualitas dan juga mengetahui kelemahan dari perangkat lunak. Tujuannya dari pengujian ini adalah untuk menjamin bahwa perangkat lunak yang memiliki kualitas yang baik yaitu

mampu untuk mempresentasikan kajian pokok dari spesifikasi, analisis, perancangan dan pengkodean dari perangkat lunak itu sendiri.

14.2.1 Rencana Pengujian

Rencana Pengujian yang akan dilakukan yaitu:

a. Pengujian halaman *costumer*

Tabel 5.1. Pengujian halaman costumer

No	Requirment yang di uji	Butir uji
1.	Halaman *costumer*	Melihat dan mencari *costumer*
2.	Halaman pembuatan *costumer* baru	Membuat data *costumer* baru
3.	Halaman *edit costumer*	*Editing* data *costumer*
4.	Halaman *detail costumer*	Melihat *detail* data *costumer*
5.	Halaman *delete costumer*	*Deleting* data *costumer*

b. Pengujian halaman *employee*

Tabel 5.2. Pengujian halaman *employee*

No	Requirment yang di uji	Butir uji
1.	Halaman *employee*	Melihat dan mencari *employee*
2.	Halaman pembuatan *employee* baru	Membuat data *employee* baru
3.	Halaman *edit employee*	*Editing* data *employee*
4.	Halaman *detail employee*	Melihat *detail* data *employee*
5.	Halaman *delete employee*	*Deleting* data *employee*

c. Pengujian halaman *item*

Tabel 5.3. Pengujian halaman *item*

No	Requirment yang di uji	Butir uji
1.	Halaman *item*	Melihat dan mencari *item*
2.	Halaman pembuatan *item* baru	Membuat data *item* baru
3.	Halaman *edit item*	*Editing* data *item*
4.	Halaman *detail item*	Melihat *detail* data *item*
5.	Halaman *delete item*	*Deleting* data *item*

d. Pengujian halaman *product*

Tabel 5.4. Pengujian halaman *product*

No	Requirment yang di uji	Butir uji
1.	Halaman *product*	Melihat dan mencari *product*
2.	Halaman pembuatan *product* baru	Membuat data *product* baru
3.	Halaman *edit product*	*Editing* data *product*
4.	Halaman *detail product*	Melihat *detail* data *product*
5.	Halaman *delete product*	*Deleting* data *product*
6.	Halaman *manage product*	*Managing item level product*

14.2.2 Kasus dan Hasil Pengujian

Berikut beberapa kasus penujian pengujian dan hasil dari pengujiannya:

1. Pengujian semua pada halaman *costumer*

Tabel 5.5. Pengujian halaman *costumer*

Requirement	Skenario uji	Hasil yang diharapkan	Hasil pengujian
Halaman Costumer	Halaman *costumer*	Akan tampil halaman *costumer*	Sesuai
	create new costumer (jika benar)	Semua data *costumer* yang diisi dengan lengkap dan memenuhi syarat, akan masuk ke database	Sesuai
	create new costumer (jika salah)	Semua data *costumer* yang diisi dengan lengkap dan memenuhi syarat, tidak akan masuk ke database	Sesuai
	Editing costumer (jika benar)	Semua data *costumer* yang telah di rubah, akan diupdate ke database	Sesuai
	Editing costumer (jika salah)	Semua data *costumer* yang telah di rubah, tidak akan diupdate ke database	Sesuai
	Melihat data *costumer*	Akan tampil data *costumer* yang di pilih	Sesuai
	Deleting costumer data	Data *costumer* akan diberi tanda saja	Sesuai

2. Pengujian semua pada halaman *employee*

Tabel 5.6. Pengujian halaman *employee*

Requirement	Skenario uji	Hasil yang diharapkan	Hasil pengujian
Halaman Employee	Halaman *employee*	Akan tampil halaman *employee*	Sesuai
	create new employee (jika benar)	Semua data *employee* yang diisi dengan lengkap dan memenuhi syarat, akan masuk ke database	Sesuai
	create new employee (jika salah)	Semua data *employee* yang diisi dengan lengkap dan memenuhi syarat, tidak akan masuk ke database	Sesuai
	Editing employee (jika benar)	Semua data *employee* yang telah di rubah, akan diupdate ke database	Sesuai
	Editing employee (jika salah)	Semua data *employee* yang telah di rubah, tidak akan diupdate ke database	Sesuai
	Melihat data *employee*	Akan tampil data *employee* yang di pilih	Sesuai
	Deleting employee data	Data *employee* akan diberi tanda saja	Sesuai

3. Pengujian semua pada halaman *item*

Tabel 5.7. Pengujian halaman *item*

Requirement	Skenario uji	Hasil yang diharapkan	Hasil pengujian
Halaman Item	Halaman *item*	Akan tampil halaman *item*	Sesuai
	create new item (jika benar)	Semua data *item* yang diisi dengan lengkap dan memenuhi syarat, akan masuk ke database	Sesuai
	create new item (jika salah)	Semua data *item* yang diisi dengan lengkap dan memenuhi syarat, tidak akan masuk ke database	Sesuai
	Editing item (jika benar)	Semua data *item* yang telah di rubah, akan diupdate ke database	Sesuai
	Editing item (jika salah)	Semua data *item* yang telah di rubah, tidak akan diupdate ke database	Sesuai
	Melihat data *item*	Akan tampil data *item* yang di pilih	Sesuai
	Deleting item data	Data *item* akan diberi tanda saja	Sesuai

4. Pengujian semua pada halaman *product*

Tabel 5.8. Pengujian pada halaman *product*

Requirement	Skenario uji	Hasil yang diharapkan	Hasil pengujian
Halaman Product	Halaman *product*	Akan tampil halaman *product*	Sesuai
	create new product (jika benar)	Semua data *product* yang diisi dengan lengkap dan memenuhi syarat, akan masuk ke database	Sesuai
	create new product (jika salah)	Semua data *product* yang diisi dengan lengkap dan memenuhi syarat, tidak akan masuk ke database	Sesuai
	Editing product (jika benar)	Semua data *product* yang telah di rubah, akan diupdate ke database	Sesuai
	Editing product (jika salah)	Semua data *product* yang telah di rubah, tidak akan diupdate ke database	Sesuai
	Melihat data *product*	Akan tampil data *product* yang di pilih	Sesuai
	Deleting product data	Data *product* akan diberi tanda saja	Sesuai
	Managing product data	Mengatur *item level product*	Sesuai

14.2.3 Kesimpulan Hasil Pengujian

Berdasarkan pada hasil pengujian kasus uji sample di atas, maka dapat ditarik kesimpulan bahwa perangkat lunak secara fungsional mengeluarkan hasil sesuai dengan yang diharapkan.