



IMPLEMENTASI *PRIVATE CLOUD* MENGGUNAKAN *RASPBERRY PI* UNTUK PENGAKSESAN DATA PRIBADI

PRIVATE CLOUD IMPLEMENTATION USING RASPBERRY PI FOR ACCESSING PERSONAL DATA

Sitti Aisa¹ dan Thabrani R²

Program Studi Teknik Informatika STMIK Dipanegara Makassar¹²

Jl. Perintis Kemerdekaan Kemerdekaan Km.9 Makassar, Tamalanrea Makassar 90000 - Indonesia

sittiaisa.11@gmail.com

Naskah Diterima: 10 November 2016; Direvisi : 07 Desember 2016; Disetujui : 22 Desember 2016

Abstrak

Penggunaan teknologi yang semakin banyak dan berkembang berdampak pada pengaksesan dan pengelolaan data dari komputer oleh pengguna pribadi menjadi tidak fleksibel karena, personal komputer membutuhkan sumber daya listrik langsung dan tempat penyimpanan yang statis. Oleh karena itu, penulis berencana mengimplementasi sebuah layanan private cloud yang menggunakan Raspberry Pi sebagai server – nya, serta akan diuji dengan pengujian Blackbox. Penelitian lakukan dengan pengumpulan data melalui studi literatur, percobaan, dan pengamatan. Metode perancangan menggunakan UML dengan use case diagram, class diagram, activity diagram, serta sequence diagram. Aplikasi ini dibangun dengan menggunakan Django, Python, RaspberryPi, MySQL. Manfaat implementasi ini dapat membantu seseorang dalam pengelolaan data dari berbagai macam sumber daya dengan cepat dan di mana saja.

Kata kunci: Private, Cloud, Aplikasi.

Abstract

The use of technology is increasing and growing that giving an impact on inflexible computer data accesing and managing by private user because personal computer requires a direct power source directly and static storage area. Therefore, the author plans to implement a private cloud service that uses raspberry pi as a server and will be tested by the test Blackbox. This study is conducted by collecting data literature study, experiment, and observation. The design method use UML use case diagram, class diagram, activity diagram, and sequence diagram. This application is built using Django, Python, Raspberry Pi, MySQL. The benefit of this implementation can help a person in data management from variety of resources in quickly and anywhere.

Keywords: Private, Cloud, Application.

PENDAHULUAN

Perkembangan teknologi khususnya dibidang teknologi informasi membuat semua pengguna membutuhkan teknologi yang cepat dan dapat diakses dari berbagai tempat dan berbagai sumber daya untuk mendukung segala jenis pekerjaan yang dilakukan sehari - hari. Teknologi informasi telah bermetamorfosis menjadi sebuah basis penting dimana hal-hal substansial dari pengguna pribadi didokumentasikan dan disimpan dalam sebuah penyimpanan data. Dalam pelaksanaannya, penyimpanan data beserta aplikasi lainnya seringkali membutuhkan sumberdaya penyimpanan *Central Processing Unit* (CPU) yang statis atau tidak dapat dibawa kemana –mana.

Dalam kondisi tersebut, sistem *Private Cloud Computing* menggunakan *Raspberry Pi* dinilai sangat bermanfaat dan berguna bagi pengguna teknologi. Hal ini disebabkan sistem *private cloud computing*, yang berupa sistem atau sumber daya yang sangat kecil, dapat diakses secara *online*.

Perkembangan dari sistem *cloud computing* dengan sistem menyewa sebuah *resource* berharga tinggi dapat dipangkas secara signifikan melalui penggunaan *private cloud computing*. *Resource* kecil, dalam hal ini *Raspberry Pi* merupakan evolusi dari beberapa *resource* di zaman dahulu yang membutuhkan tempat yang besar. *Raspberry Pi* juga merupakan salah satu *resource* CPU yang sangat *powerfull* untuk membuat sebuah *server* kecil karena dukungan sistem operasi yang juga sangat terkenal di kalangan *server* yakni *GNU Linux*.

Dari pemaparan diatas, rumusan masalah penelitian ini adalah bagaimana mengimplementasikan *private cloud* yang dapat digunakan oleh pengguna teknologi sekarang ini

untuk mengakses data-data pribadi mereka dimana saja dan kapan saja.

Tujuan penelitian ini mengimplementasikan *private cloud* menggunakan *Raspberry Pi*, sehingga dapat memberikan kemudahan bagi pengguna pribadi untuk mengakses data dari mana saja dan dapat menjalankan aplikasi basis data sederhana dari *Raspberry Pi* yang dijadikan *server* mini.

Ada beberapa penelitian yang sebelumnya dilakukan mengenai teknologi *cloud computing* atau *cloud storage*. Dalam upaya mengembangkan dan menyempurnakan pengembangan *cloud computing* ini, perlu dilakukan studi pustaka (*literature review*) sebagai salah satu dari penerapan metode penelitian yang dilakukan, diantaranya :

1. Penelitian yang dilakukan Irawan(2013). Sistem tersebut diusulkan untuk menunjukkan bagaimana merancang sistem penyimpanan elektronik dengan menggunakan teknologi berbasis *cloud computing* dan validasi MD5, sehingga tidak sembarang orang dapat mengakses sistem tersebut.
2. Penelitian yang telah dilakukan oleh Budi (2013). Sistem tersebut diusulkan untuk membuat dan mengimplementasikan teknologi berbasis *cloud computing* yang dapat dijadikan sebagai *cluster* komputer untuk *web server*.
3. Penelitian yang telah dilakukan oleh Natsirudin (2011). Sistem ini digunakan untuk menganalisis sebuah pemanfaatan teknologi *cloud computing* yang bersifat fleksibel pada jaringan *thin client*.
4. Penelitian yang telah dilakukan oleh Harsono (2010). Sistem ini diusulkan untuk menganalisis dua sistem teknologi yang

digunakan *user* pada saat ini dan yang akan datang, sehingga dapat dimanfaatkan sebagai teknologi yang efektif dan efisien.

5. Penelitian yang telah dilakukan oleh Armanda (2010). Sistem tersebut diusulkan untuk mengimplementasikan pemanfaatan teknologi berbasis *cloud computing*, sehingga dapat digunakan secara lebih efektif, efisien, dan fleksibel.

Dari beberapa sumber *literature review* maka peneliti dapat mengetahui bahwa penelitian tentang *cloud computing* menggunakan *Raspberry Pi* untuk pengaksesan data pribadi belum pernah dilakukan.

Cloud Computing

Cloud computing adalah sebuah model yang memungkinkan adanya penggunaan sumber daya (*resource*) secara bersama-sama dan mudah, menyediakan jaringan akses di mana-mana, dapat dikonfigurasi, dan layanan yang digunakan sesuaikeperluan (*on demand*). (Pratama, 2014).

Pada teknologi *cloud computing*, terdapat tiga model layanan yang dapat dipilih sesuai kebutuhan. Ketiga model layanan yang disajikan tersebut meliputi *Infrastructure as a Service* (IAAS), *Platform as a Service* (PAAS), dan *Software as a Service* (SAAS). Pembagian jenis layanan ini dilakukan dengan tujuan untuk membantu dalam menyesuaikan keperluan pengguna, sehingga pengguna dapat terbantu dengan adanya layanan yang disajikan tersebut. (Pratama, 2014).

IAAS atau *Cloud IAAS* merupakan jenis layanan pada *cloud computing* yang menekankan layanan penyediaan sarana jaringan komputer (*computer network*), perangkat keras jaringan, komputer *server*, media penyimpanan (*storage*), *processor*, beserta proses virtualisasi, yang menunjang proses komputasi.

PAAS atau *Cloud PAAS* merupakan jenis layanan menekankan penyediaan *platform* untuk membantu proses pengembangan perangkat lunak secara cepat dan mudah. Layanan *platform* yang digunakan umumnya juga berbasis *web*, di mana di dalamnya telah disediakan banyak fitur yang memudahkan *programmer* dan pengguna awam di dalam mengembangkan aplikasi tanpa memerlukan banyak proses penulisan sumber kode (*coding*).

SAAS merupakan jenis layanan yang diberikan oleh teknologi *cloud computing* kepada para penggunanya dalam bentuk pemakaian bersama perangkat lunak (aplikasi). Umumnya layanan SAAS disediakan dalam bentuk tatap muka berbasis *web*.

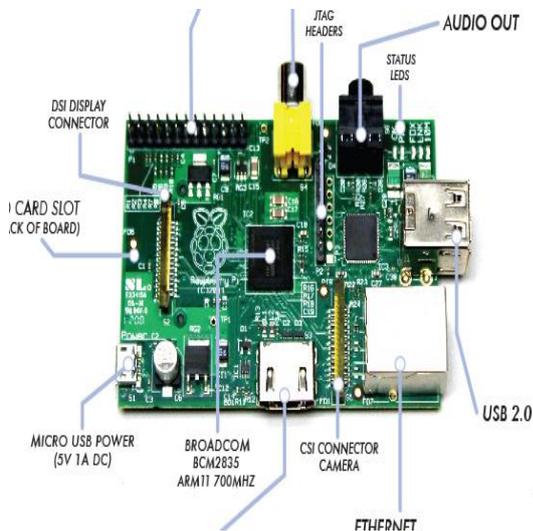
Model Deployment Cloud Computing

Model *deployment cloud computing* dibagi berdasarkan lingkungan, kondisi, dan keperluan pengguna sehingga teknologi *cloud computing* dapat dimanfaatkan dengan baik dan mudah. Adapun model *deployment cloud computing* ialah sebagai berikut :

1. *Private cloud* ditujukan untuk penggunaan terbatas pada kalangan tertentu saja (*private*). Model ini umumnya diterapkan pada lingkungan sekolah, laboratorium, atau bahkan pribadi.
2. *Public cloud* biasanya diletakkan dilokasi publik (misalnya di jaringan internet dan memiliki IP *public*), sehingga layanan data dan informasi di dalamnya dapat digunakan oleh semua pengguna.
3. *Comunity cloud* merupakan layanan *cloud computing* yang dibangun oleh satu atau beberapa komunitas.
4. *Hybrid cloud* merupakan gabungan *private cloud* dan *public cloud* sehingga di dalamnya terdapat manajemen keamanan dan manajemen data.

Raspberry Pi

Raspberry Pi adalah sebuah processor *Advanced RISC Machine/Acorn RISC Machine (ARM)* yang sangat kecil dan murah, yang dapat menjalankan sistem operasi berbasis *Linux* atau sistem operasi ringan lainnya. *Raspberry Pi* ini sudah memiliki beberapa port seperti port *High – Definition Multimedia Interface (HDMI)*, port *Radio Corporation of America (RCA) video*, port audio, port *ethenet*, dan port *Universal Serial Bus (USB)* 2.0. Lihat pada Gambar 1 untuk tampilan *Raspberry Pi*.

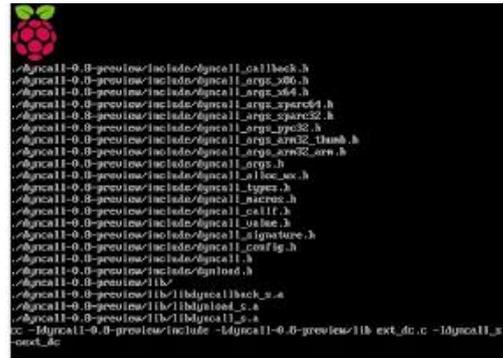


Gambar 1 :Raspberry Pi Model B (512MB RAM)
Sumber: (Kiepert, 2013)

Instalasi Sistem Operasi Raspbian

Raspbian adalah sistem operasi lengkap berbasis *Linux*, tersedia secara bebas dan mempunyai dukungan, baik yang berasal dari komunitas maupun tenaga ahli profesional. *Raspbian* merupakan sistem operasi turunan dari *distro Linux Debian*. Dengan sistem operasi berbasis *open source* dan berbasis *Debian*, *Raspbian* semakin diminati oleh para pengguna khususnya sebagai sistem operasi untuk *Raspberry Pi*. Beberapa langkah instalasi *Raspbian* pada *Raspberry Pi* (Suranata, 2015):

1. Masukkan *SD card* pada *Raspberry Pi* dan hidupkan

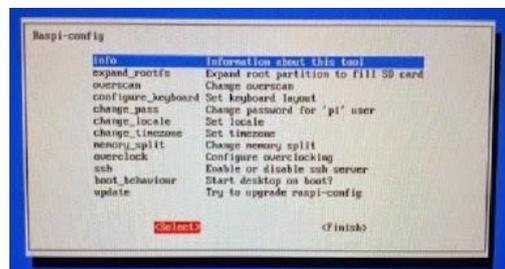


2. Login *Raspberry pi*



Gambar 3. proseslogin pada *Raspberry Pi*

3. Konfigurasi *Raspberry Pi*



Gambar 4. proses konfigurasi pada *Raspberry Pi*

Bahasa Pemrograman Python

Berdasarkan *official web Python* <https://wiki.python.org/moin/>, *Python* adalah bahasa pemrograman interpretatif multiguna dengan perancangan berfokus pada tingkat keterbacaan kode. *Python* diklaim sebagai bahasa yang menggabungkan kapabilitas, kemampuan sintaksis kode yang sangat jelas, dilengkapi fungsionalitas pustaka standar yang besar serta komprehensif.

Beberapa kelebihan dari bahasa pemrograman *Python* adalah sebagai berikut :

1. Memiliki kepustakaan yang luas. Dalam distribusi *Python* disediakan modul-modul yang dapat dipakai untuk berbagai keperluan.
2. Memiliki tata bahasa yang jernih dan mudah dipelajari.
3. Memiliki aturan *layout* kode sumber yang memudahkan pengecekan, pembacaan kembali, dan penulisan ulang kode sumber.
4. Merupakan bahasa yang telah berorientasi objek.
5. Memiliki sistem pengolahan memori otomatis (*garbage collection* seperti *Java*)

Modul mudah dikembangkan dengan menciptakan modul-modul baru. Modul-modul tersebut dapat dibangun dengan bahasa *Python* dan *C/C++*.

Django

Berdasarkan *official web Django* <https://www.djangoproject.com/>, *Django* adalah sebuah *web framework* yang *open source* dan berbasis *Python*. *Web framework* adalah alat yang mempermudah dalam membangun suatu *website*. *Django* menggunakan sistem *Model Template and View (MTV)* sehingga memudahkan dalam membangun sebuah *website*. *Model* adalah *layer* yang digunakan untuk berinteraksi dengan *database*. *Template* adalah *layer* yang digunakan untuk menangani masalah tampilan seperti *XML*, *HTML*, dan lainnya. Sedangkan *view* adalah *layer* yang menghubungkan *layer model* dan *template* yang di dalamnya berisikan logika pengolah data dari *model* dan menampilkannya di *template*. *Django* memiliki kelebihan diantara *web framework* lainnya yaitu :

1. *Object-Relational Mapper*: mendefinisikan data model dalam *Python* dan menggunakan

Application Programming Interface (API) untuk mengakses data tersebut.

2. *Automatic Admin Interface Django*: menyediakan *admin interface* secara otomatis, sehingga tidak perlu repot untuk membuat *Back-End website*.
3. *Elegant Uniform Resource Locator (URL) Design*: pembuatan URL yang lebih mudah dan fleksibel.
4. *Template System*: Sistem *template Django* merupakan salah satu yang paling baik dibandingkan dengan *web framework* lainnya.
5. *Cache System*: dapat menggunakan sistem *cache* dengan mudah.

MySQL

MySQL sebagai salah satu jenis *database server* yang sangat terkenal. Kepopulerannya disebabkan *MySQL* menggunakan *SQL* sebagai bahasa dasar untuk mengakses *database* – nya. Selain itu, ia bersifat *open source* pada berbagai platform kecuali untuk jenis *enterprise* yang bersifat komersial. *MySQL* sebenarnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya, yaitu *SQL (Structured Query Language)*. *SQL* adalah sebuah konsep pengoperasian basisdata, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. (Kadir, 2008)

MySQL termasuk jenis *RDBMS (Relation Database management Sistem)*. Itulah sebabnya istilah seperti *table*, baris, dan kolom digunakan pada *MySQL*. Pada *MySQL* sebuah *database* mengandung salah satu atau sejumlah *table*. *Table* terdiri atas sejumlah baris dan setiap baris mengandung satu atau beberapa kolom terstruktur. (Kadir, 2008)

MySQL dapat di kelompokkan menjadi 3 macam yaitu:

1. DDL (*Data Definition Language*)

DDL bertugas untuk membuat objek SQL dan menyimpan definisi ini dalam *table*. Contoh dari objek yang dimaksud diatas adalah *table*, *view*, dan *index*.Pembuat *table*, perubahan struktur *table*, perubahan nama *table* serta perintah-perintah untuk menghapus *table* dilakukan dengan sub bahasa yang tergolong dalam DDL, yaitu *create*, *alter*, dan *drop*.

2. DML (*Data Manipulation Language*)

DML digunakan untuk memproses data dalam objek skema.Dengan menggunakan perintah-perintah ini dapat menampilkan data (*select*), mengubah data (*update*), menghapus data (*delete*), dan menambahkan atau menyisipkan data baru (*insert*).

3. DCL (*Data Control language*)

Sebagai alat kontrol keamanan terhadap database dan tabelnya, terdapat dua perintah utama yaitu *grant* dan *revoke*. *Grant* digunakan untuk mengizinkan user mengakses *table* dalam *database* tertentu, sedangkan *revoke* adalah sebaliknya. (Kadir, 2008)

METODE

Jenis Penelitian

Jenis penelitian adalah kepustakaan, karena tidak ada tempat/objek penelitian. Dilakukan dengan mencari referensi dari buku-buku dan sumber-sumber yang berasal dari jurnal di internet, sehingga dapat mendukung penelitian dan diterapkan pada aplikasi yang dirancang.

Teknik Pengumpulan Data

Pengumpulan data merupakan metode yang difungsikan untuk memperoleh informasi-informasi atau data-data terhadap kasus yang menjadi permasalahan dalam laporan tugas akhir ini. Hal yang dibutuhkan oleh penulis adalah informasi-informasi mengenai metode dan rancangan yang digunakan dalam penelitian kasus ini. Ada dua pendekatan yang digunakan untuk memperoleh informasi-informasi ini, di antaranya :

1. Studi *literature*

Berupa pencarian sumber-sumber bacaan yang dapat menunjang topik dan sebagai landasan teoritis yang lebih meyakinkan. Sumber bacaan yang dapat menjadi referensi tersebut berupa *text book*, tugas akhir, buku panduan belajar pemrograman, maupun sumber bacaan *softcopy* yang diperoleh dari media internet.

2. Percobaan dan pengamatan

Melakukan percobaan pada aplikasi melalui *Raspberry Pi* dan secara periodik dalam sinkronisasi aplikasi.

Tahapan Perancangan

Tahap-tahap yang dilakukan dalam perancangan sistem ini adalah:

1. Analisis masalah : menganalisis kekurangan sistem yang sedang berjalan.
2. Pengumpulan data : mengumpulkan informasi yang dilakukan secara langsung melalui studi literatur.
3. Analisis sistem : penguraian dari suatu aplikasi yang utuh, mengidentifikasi, dan mengevaluasi permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikannya.
4. Perancangan aplikasi : merupakan strategi untuk memecahkan masalah dan

mengembangkan solusi terbaik bagi permasalahan.

5. Pengujian program : mengetahui cara kerja dari aplikasi yang dirancang secara terperinci sesuai spesifikasi dan menilai apakah setiap fungsi atau prosedur yang dirancang sudah bebas dari kesalahan logika.
6. Implementasi : tahap dimana aplikasi siap untuk diterapkan, maka pada kegiatan ini dilakukan pengetesan secara langsung dengan pemakai atau *user* pada periode tertentu, bila pada kegiatan ini ternyata sistem sudah berjalan baik maka dinyatakan dapat digunakan.

Alat Dan Bahan Penelitian

Dalam proses perancangan ini, diperlukan alat dan bahan yang dapat mendukung keberhasilan perancangan. Alat dan bahan perancangan yang digunakan adalah :

Alat

1. *Hardware*, terdiri dari :
 - a. *Raspberry Pi* model B dengan spesifikasi :
 - 1) *Processor ARM 11(700 MHz)*
 - 2) *Memori RAM 512*
 - 3) *External memory 8 GB*
 - b. Laptop Asus dengan spesifikasi :
 - 1) *Processor AMD Dual Core*
 - 2) *Memory RAM 2 GB*
 - 3) *Harddisk 320 GB*
 - 4) *ATI Radeon Graphic*
2. *Software*, terdiri dari :
 - a. Sistem Operasi *Raspbian Whezzy*
 - b. *Python 2.7*

c. *Django* versi 15

d. *MySQL*

Teknik Pengujian Sistem

Model pengujian yang dilakukan dalam penelitian ini adalah *black box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian *black box* memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Pengujian *black box* berusaha menemukan kesalahan dalam kategori sebagai berikut:

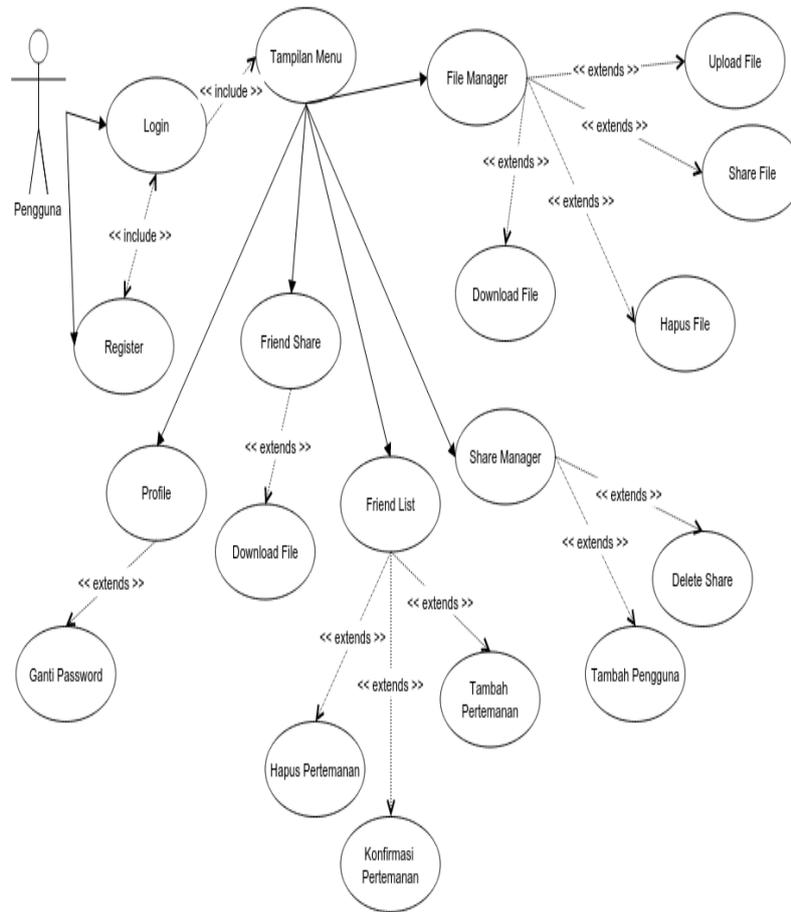
1. Fungsi-fungsi yang tidak benar atau hilang.
2. Kesalahan *interface*.
3. Kesalahan dalam struktur data atau akses *database* eksternal.
4. Kesalahan kinerja.
5. Inisialisasi dan kesalahan terminasi.

Bahan

Bahan yang digunakan dalam penelitian ini adalah basis data sederhana.

Perancangan Sistem

Perancangan aplikasi *private cloud* merupakan suatu kegiatan yang dilakukan untuk mendesain suatu aplikasi yang mempunyai algoritma yang tersusun secara logis, dimulai dari pengumpulan data yang diperlukan untuk pelaksanaan perancangan tersebut sampai pada perancangannya.

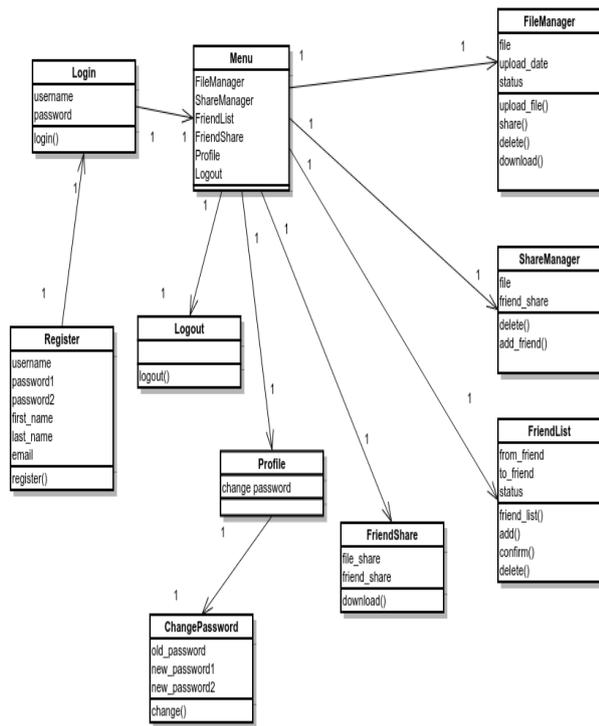


Gambar 5: Use Case Diagram

Gambar 5 diatas menjelaskan bahwa pada aplikasi ini, actor menggunakan aplikasi dengan cara login atau registrasi terlebih dahulu kemudian tampil halaman *File Manager* yang di dalamnya terdapat

menu *File Manager*, menu *Share Manager*, menu *Friend List*, menu *Friend Share*, *Profile*, dan *Logout*. (Suhendar. & Hariman, 2008)

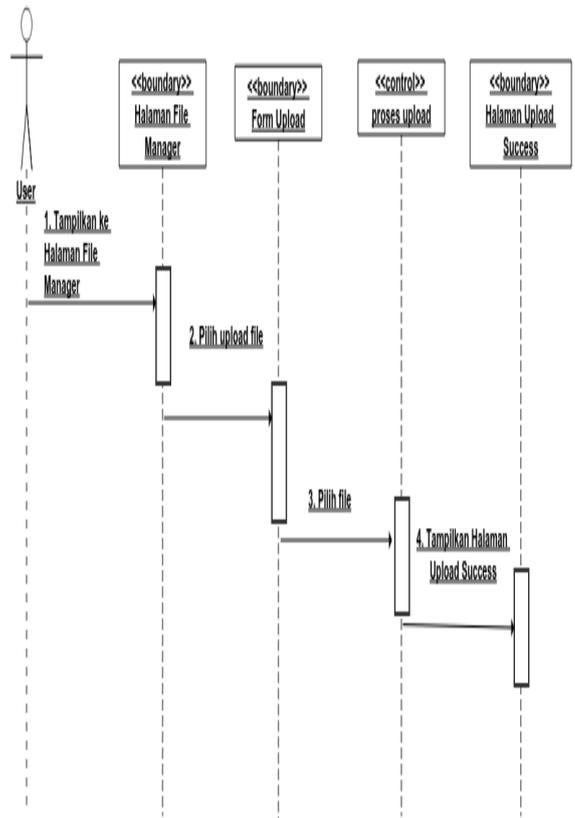
Class Diagram



Gambar 6 :Class Diagram

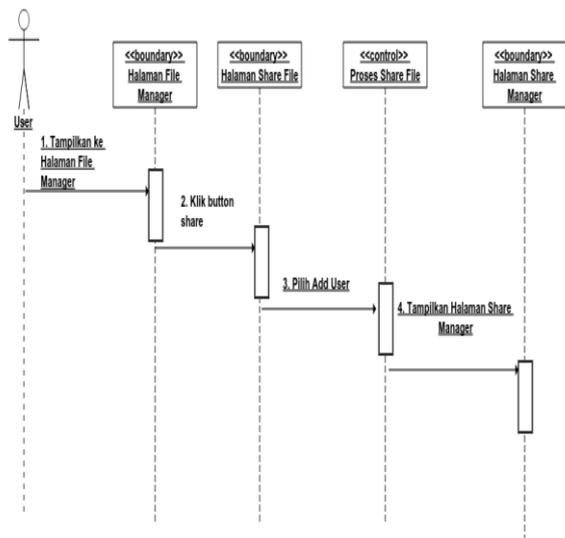
Gambar 6 memberikan pandangan secara luas dari suatu sistem dengan menunjukkan kelas-kelasnya dan hubungan mereka. Aplikasi ini memiliki class yang terdiri dari 10 class yaitu class *Login*, class *Register*, class *Menu* yang terdiri dari class *File Manager*, class *ShareManager*, class *FriendList*, class *FriendShare*, class *Profile*, dan class *Logout*. Class *Profile* terdiri dari class *Change Password*.

Sequence Diagram



Gambar 7 :Sequence Diagram Upload File

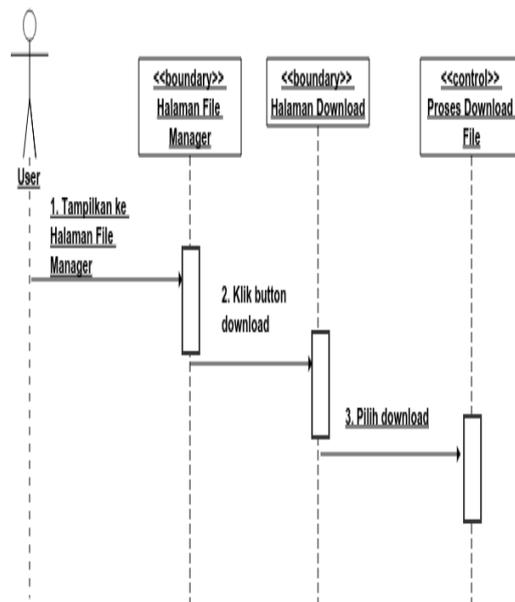
Gambar 7 menjelaskan proses yang berlangsung pada *activity diagram* melakukan *upload file*, pada halaman *File Manager* meng-klik tombol *Upload File* kemudian tampil *form Upload*. Pada *form Upload*, user meng-klik *Choose File* lalu memilih *file* yang akan di – *upload* lalu meng-klik tombol *Upload* (Suhendar. & Hariman, 2008)



Gambar 8 :Sequence Diagram Share File

Gambar 8 menjelaskan tentang proses yang berlangsung pada *activity diagram* melakukan *share file*, pada halaman *File Manager* meng-klik tombol *icon Share* yang terletak pada kolom *action*. Lalu akan muncul halaman *File Share*, Klik tombol *Add* untuk menambahkan user yang dapat mengakses *file*.

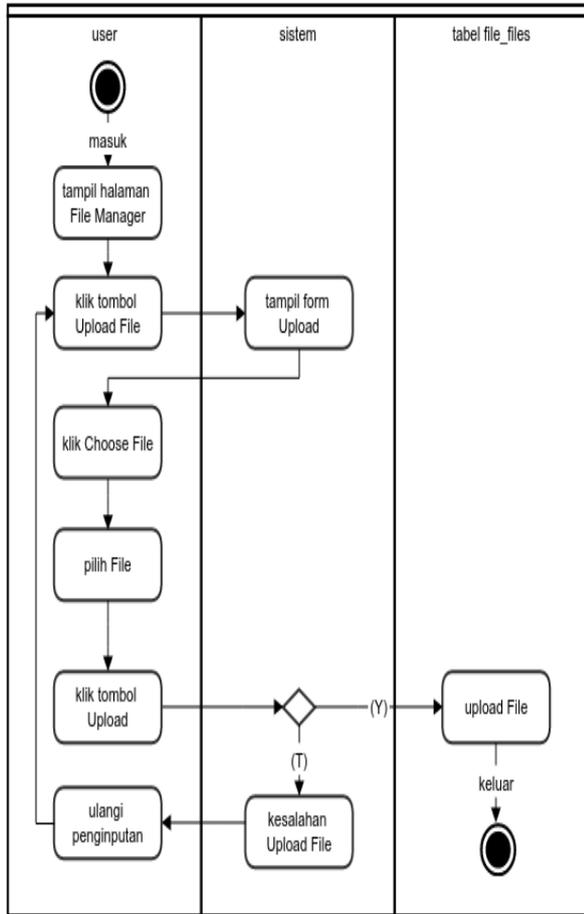
Sedangkan pada gambar 9 menjelaskan tentang proses yang berlangsung pada *activity diagram* melakukan *download file*, pada halaman *file manager* meng-klik tombol *icon download* yang terletak pada kolom *action*. Lalu akan muncul halaman *Download*, Klik tombol *Download* untuk dapat mengunduh *file*. (Suhendar. & Hariman, 2008)



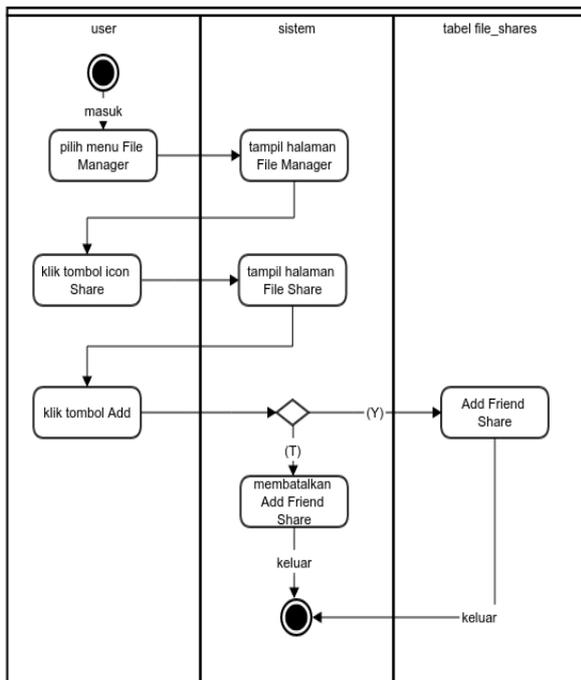
Gambar 9 :Sequence Diagram Download File **Activity Diagram**

Untuk *activity diagram* yang di jelaskan dibawah ini adalah *activity diagram upload file*, *activity diagram share file* dan *activity diagram download file*.

Pada gambar 10dibawah ini dijelaskan proses yang berlangsung pada *activity diagram* melakukan *upload file*, pada halaman *file manager* meng-klik tombol *Upload File* kemudian tampil *form upload*. Pada *form upload*, *user* meng-klik *choose file* lalu memilih *file* yang akan diupload lalu meng-klik tombol *Upload*. (Suhendar. & Hariman, 2008)

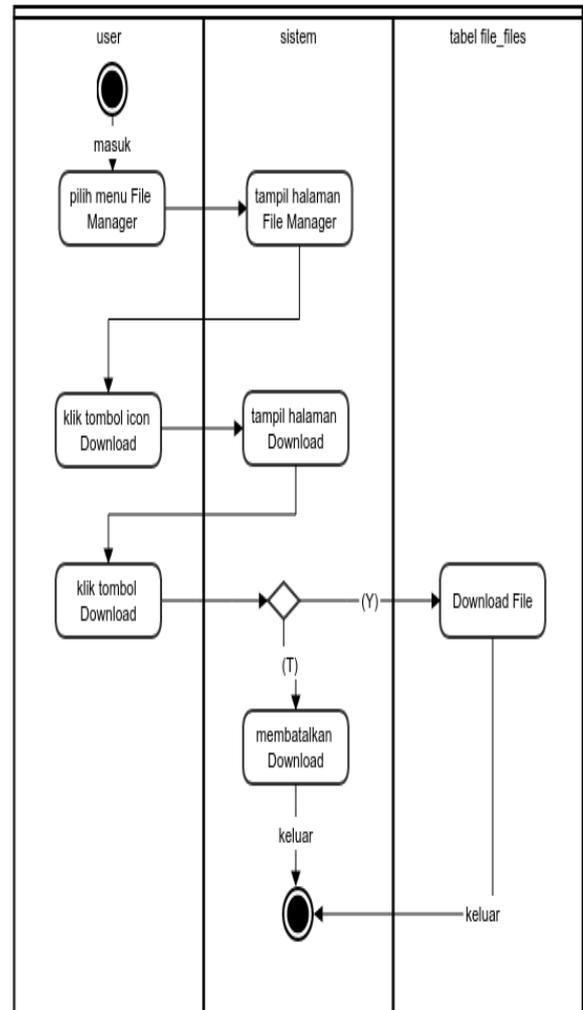


Gambar 10: Activity Diagram Upload File



Gambar 11 :Activity Diagram Share File

Proses yang berlangsung pada *activity* diagram Gambar 11 menjelaskan tentang melakukan *share file*, pada halaman *file manager* meng-klik tombol *icon share* yang terletak pada kolom *action*. Lalu akan muncul halaman *File Share*, Klik tombol *Add* untuk menambahkan user yang dapat mengakses *file*. (Suhendar. & Hariman, 2008)



Gambar 12: Activity Diagram Download File

Gambar 12 menjelaskan tentang proses yang berlangsung pada *activity* diagram melakukan *download file*, pada halaman *file manager* meng-klik tombol *icon download* yang terletak pada kolom *action*. Lalu akan muncul halaman *Download*, Klik tombol *Download* untuk dapat mengunduh *file*. (Suhendar. & Hariman, 2008)

HASIL DAN PEMBAHASAN

Pada bagian ini, di jelaskan secara detail tampilan aplikasi dan proses yang harus dilakukan dalam menjalankan aplikasi ini sampai selesai. Berikut ini tampilan aplikasinya :

a. Form Sign Up

Gambar 13 :Form Sign Up

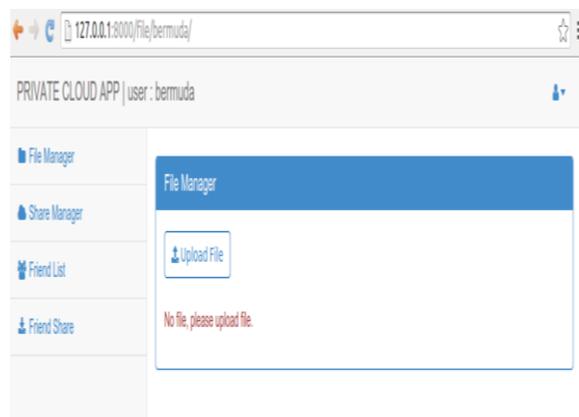
Pada gambar 13 di perlihatkan *form sign up* yang bisa digunakan untuk mendaftarkan user dan password yang digunakan untuk bisa masuk kedalam sistem tersebut.

b. Form Login

Gambar 14. Form Login

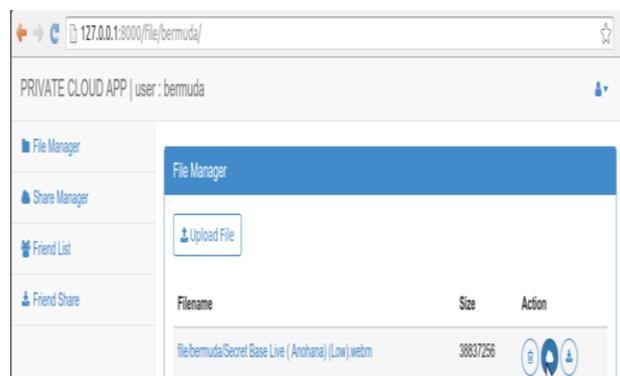
Pada gambar 14 di perlihatkan *form login* untuk pengguna yang telah mendaftarkan pada *form sign up*. Pada form ini, pengguna bisa melakukan *login* untuk masuk ke menu utama dari sistem ini. Pada gambar 15 diperlihatkan ketika pengguna

telah melakukan *login*, sistem ini secara otomatis langsung mengarahkan pada menu file manager. maka akan menampilkan semua file yang telah diunggah user tertentu yang telah login pada aplikasi. (Supriyanto, 2007)



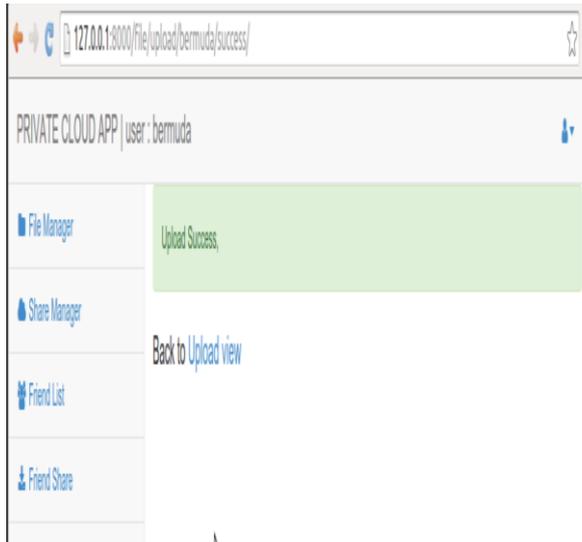
Gambar 15 :Form File Manager

c. Form Upload File



Gambar 16. Form untuk upload File sebelum file di masukkan

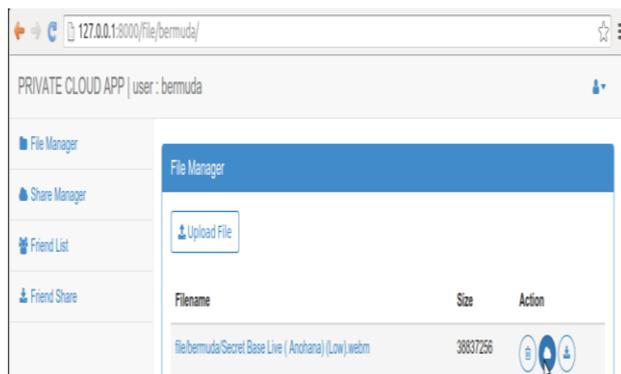
Pada gambar 16 di perlihatkan ketika tombol *Upload File* dipilih maka secara otomatis menampilkan halaman *upload file*. (Supriyanto, 2007)



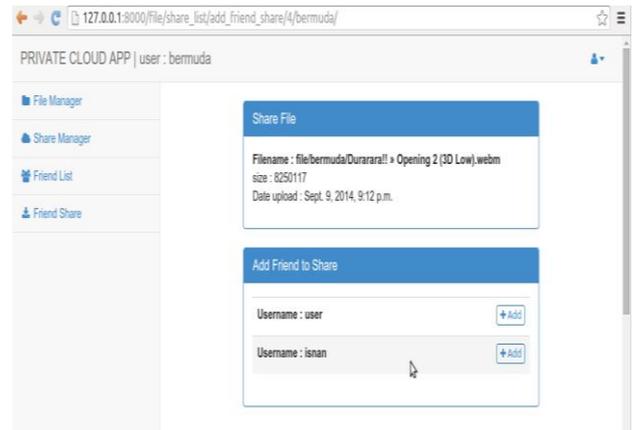
Gambar 17. *Form upload File* setelah file sukses

Pada gambar 17 di perhatikan ketika menekan tombol *Browse* maka otomatis muncul *file* dari komputer. Begitu juga dengan tombol *Upload*, maka secara otomatis *file* akan terupload dan muncul informasi bahwa *upload* telah sukses. (Supriyanto, 2007)

d. Form Share File pada File Manager



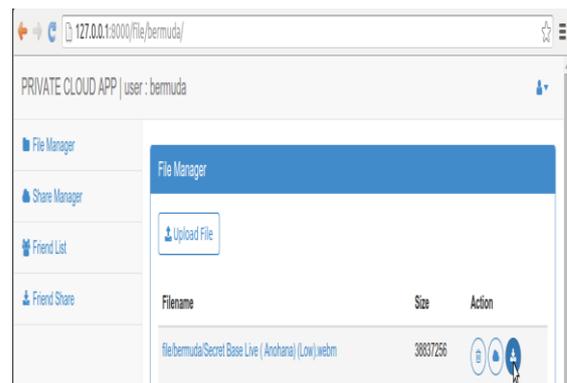
Gambar 18. *Form Share File* pada *File manager*



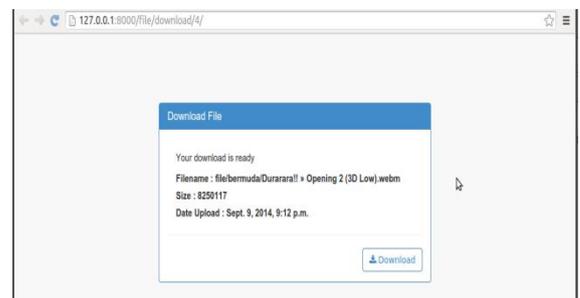
Gambar 19. *Form Share File*

Pada gambar 18 dan 19 terlihat pada saat tombol *icon share* dipilih, maka akan menampilkan halaman *file share*. (Supriyanto, 2007)

e. Form File Manager untuk icon download



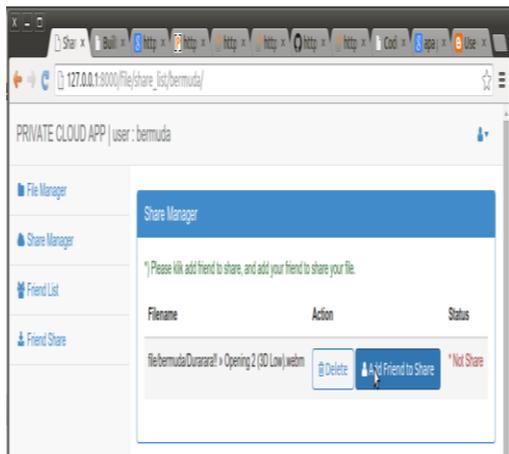
Gambar 20. *Form File Manager* untuk *icon download*



Gambar 21. *Form Download File*

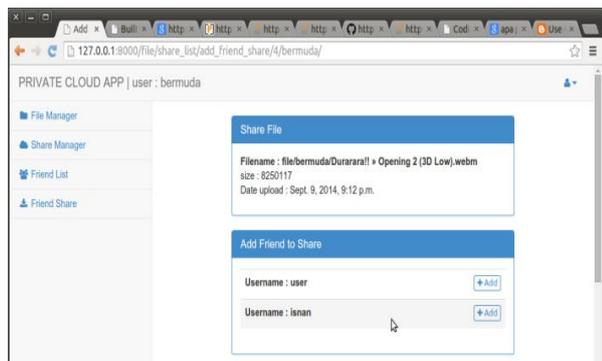
Pada gambar 20 di perhatikan *form file manager* dimana untuk mendownload *file* memilih *icon download*. Setelah itu dimunculkan pada gambar 21 *form download file* dan tekan *bottom download*.

f. Form Share Manager



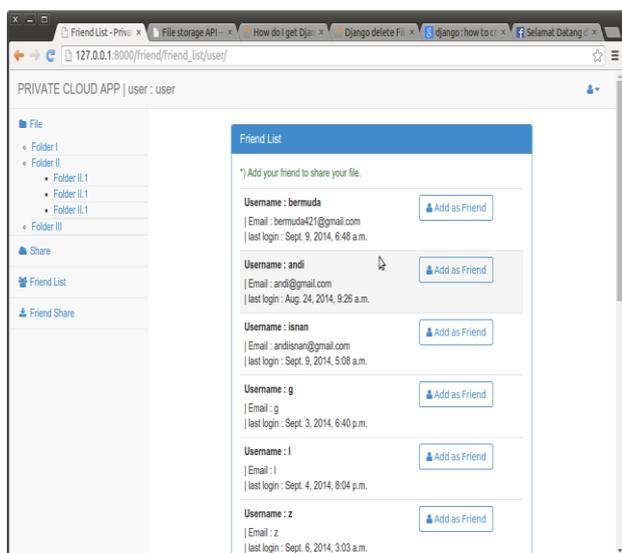
Gambar 22Form Share Manager

Pada gambar 22 terlihat pada saat tombol icon *Add Friend to Share* dipilih, maka akan menampilkan halaman *file share*.



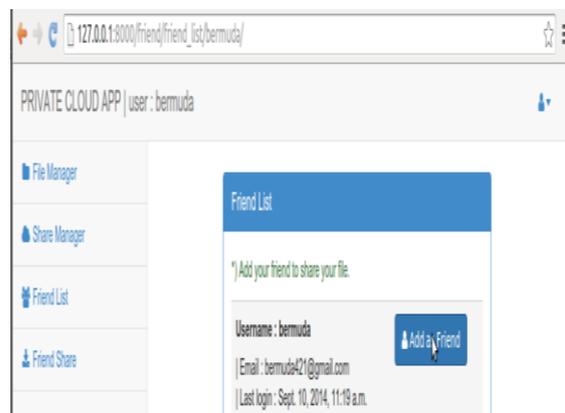
Gambar 23.Form Share File

g. Form Friend List



Gambar 24.Form Menu Friend List

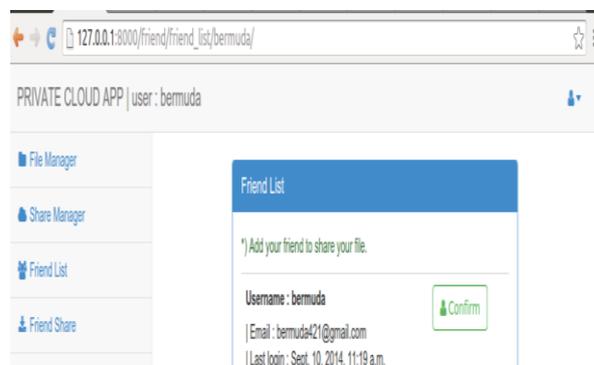
Pada gambar 24 terlihat pada saat memilih *menu friend list* maka akan menampilkan semua user yang menggunakan aplikasi.



Gambar 25.Form Friend List untuk Add friend

Pada gambar 25 terlihat pada saat tombol *add as friend* dipilih, maka akan sistem akan mengirim permintaan pertemanan dan menampilkan halaman *friend list* kembali. (Supriyanto, 2007)

h. Form Friend List untuk konfirmasi



Gambar 26.Form Friend List untuk konfirmasi

Pada gambar 26 terlihat pada saat tombol *confirm* dipilih, maka akan sistem akan mengkonfirmasi permintaan pertemanan dan menampilkan halaman *friend list* kembali

PENUTUP

Implementasi aplikasi *private cloud* dengan *raspberry pi* telah selesai dan bisa digunakan mengakses data pribadi oleh masyarakat dengan berbagai macam fitur yang ada didalamnya seperti *share file*, *upload file* dan *download file* dari siapa saja dengan mudah dan cepat.

Untuk pengembangan dari penelitian selanjutnya bisa menambahkan fitur untuk menentukan jenis *file* yang bisa di *upload* dan di *download* dalam *file manager*.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada pihak yang telah memberikan bantuan baik dari segi penyelesaian serta implementasi dari aplikasi ini. Sehingga penelitian ini dapat di sajikan dalam bentuk jurnal

DAFTAR PUSTAKA

- Kadir, A. (2008). *Dasar Perancangan dan Implementasi Database Relasional*. Yogyakarta: Andi Offset.
- Kiepert, J. (2013). *Creating a Rasperry Pi-Based Beowulf Cluster*. Boise State University.
- Pratama, I. P. A. E. (2014). *Smart City Beserta Cloud Computing dan Teknologi-teknologi Pendukung Lainnya*. Bandung: Informatika Bandung.
- Suhendar., & Hariman, G. (2008). *Visual Modelling Menggunakan UML dan Rational Rose*. Jakarta: Informatika.
- Supriyanto, A. (2007). *Web dengan HTML & XML*. Yogyakarta: Graha Ilmu.
- Suranata, A. (2015). Ulasan cara install 6 sistem operasi resmi yang dapat digunakan untuk rasperry pi.

