

**LAYANAN PENYIMPANAN BERBASIS *PRIVATE CLOUD*
(STUDI KASUS JURUSAN INFORMATIKA UNIVERSITAS ISLAM
INDONESIA)**

TUGAS AKHIR

**Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana
Strata-1 Teknik Informatika**



Oleh :

Nama : Nashihun Amien

No. Mahasiswa : 09 523 029

JURUSAN TEKNIK INFORMATIKA

FAKULTAS TEKNOLOGI INDUSTRI

UNIVERSITAS ISLAM INDONESIA

YOGYAKARTA

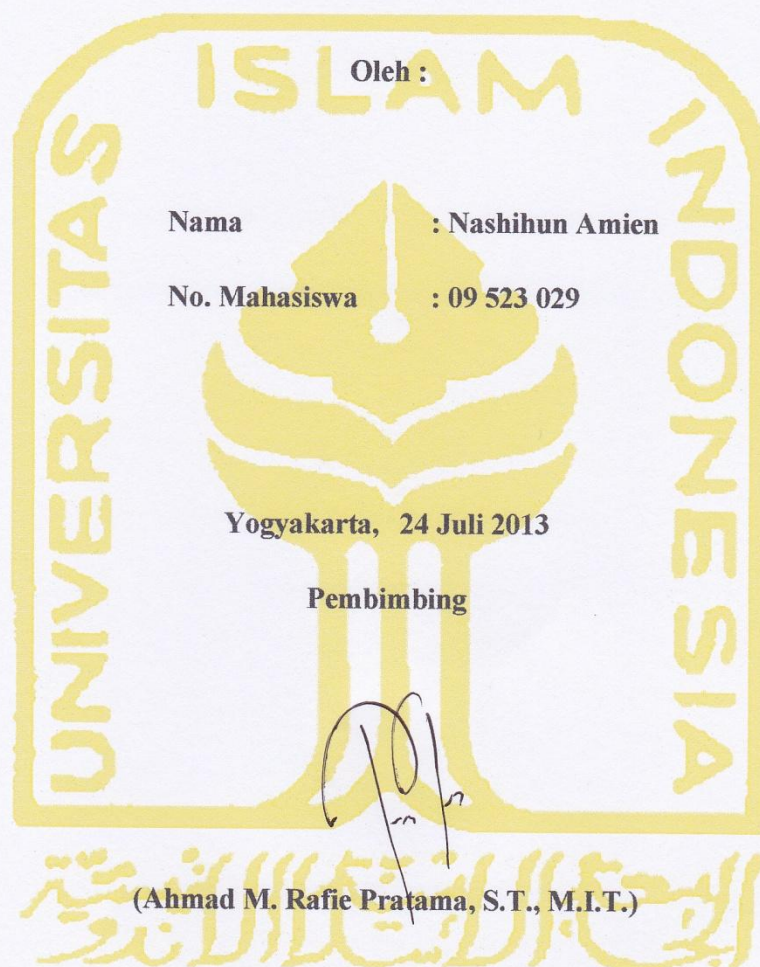
2013

LEMBAR PENGESAHAN DOSEN PEMBIMBING

LAYANAN PENYIMPANAN BERBASIS *PRIVATE CLOUD*

**(STUDI KASUS JURUSAN INFORMATIKA UNIVERSITAS ISLAM
INDONESIA)**

TUGAS AKHIR



LEMBAR PERNYATAAN KEASLIAN HASIL TUGAS AKHIR

Saya yang bertandatangan di bawah ini,

Nama : Nashihun Amien

NIM : 09 523 029

Menyatakan bahwa seluruh komponen dan isi dalam Laporan Tugas Akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti bahwa ada beberapa bagian dari karya ini adalah bukan hasil karya saya sendiri, maka saya siap menanggung risiko dan konsekuensi apapun.

Demikian pernyataan ini saya buat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 24 Juli 2012

Penulis

Nashihun Amien

LEMBAR PENGESAHAN DOSEN PENGUJI
LAYANAN PENYIMPANAN BERBASIS *PRIVATE CLOUD*
(STUDI KASUS JURUSAN INFORMATIKA UNIVERSITAS ISLAM
INDONESIA)

TUGAS AKHIR

Oleh :

Nama : Nashihun Amien

No. Mahasiswa : 09 523 029

Telah Dipertahankan di Depan Sidang Penguji sebagai Salah Satu Syarat
untuk Memperoleh Gelar Sarjana Jurusan Teknik Informatika Fakultas
Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 24 Juli 2013

Tim Penguji

Ahmad M. Raf'ie Pratama, ST., MIT.

Ketua

Yusuf Yudi Prayudi, S.Si., M.Kom.

Penguji I

R. Teduh Dirgahayu, ST., M.Sc., Ph.D

Penguji II

Mengetahui,

Ketua Jurusan Teknik Informatika

Fakultas Teknologi Industri

Universitas Islam Indonesia



Yudi Prayudi, S.Si., M.Kom.

HALAMAN PERSEMBAHAN

Dengan penuh kasih kupersembahkan untuk...

Ibu dan Bapak terkasih yang selalu mendukung, disetiap langkah, selalu memberi kasih sayang terhadap anak-anaknya. Semoga kelak aku bisa membalas semua kasih sayangmu.

Adik tersayang yang selalu mendukung dan menemani setiap langkah untuk menjadi lebih dewasa.

Sahabat, teman dan orang-orang yang selalu mendukung dari awal hingga akhir perjuangan pengerjaan tugas akhir ini.

MOTTO

“Orang besar bukan orang yang otaknya sempurna tetapi orang yang mengambil sebaik-baiknya dari otak yang tidak sempurna” - Muhammad Sallallahu Alaihi wa Sallam

“Barangsiapa mengharap perjumpaan dengan Tuhannya, maka hendaklah ia mengerjakan amal yang saleh dan janganlah ia mempersekutukan seorangpun dalam beribadat kepada Tuhannya.” - QS Al Kahfi : 110

“Sesungguhnya Allah tidak akan mengubah nasib suatu kaum sampai mereka sendiri mengubah dirinya” - QS. Ar Ra’d:11

“VI VERI VENIVERSUM VIVUS VICI. By the Power of Truth, I, while living, have Conquered the Universe.” – Alan Moore

“Never tickle a sleeping dragon.” – J.K. Rowling

“Live as if you were to die tomorrow. Learn as if you were to live forever.” – Mahatma Gandhi

KATA PENGANTAR



Assalamu'alaikum warahmatullahi wabarakatuh

Alhamdulillah, dengan mengucapkan syukur kehadiran Allah SWT yang senantiasa memberikan petunjuk serta melimpahkan berkah dan rahmat-Nya, sehingga Tugas Akhir ini dapat diselesaikan dengan judul **“LAYANAN PENYIMPANAN BERBASIS *PRIVATE CLOUD* (STUDI KASUS JURUSAN INFORMATIKA UNIVERSITAS ISLAM INDONESIA)”**.

Ucapan terimakasih kami ucapkan kepada segenap pihak yang secara langsung maupun tidak langsung telah membantu serta memudahkan proses terselesaikannya Tugas Akhir ini. Semoga Allah SWT melimpahkan rahmat serta memberikan balasan yang sebaik-baiknya untuk semua. Ucapan terima kasih saya tujukan kepada:

1. Allah Subhanahu Wa Ta'ala yang selalu melimpahkan rahmat dan hidayah-Nya kepada penulis.
2. Kedua orang tuaku tercinta Ibunda (Siti Suwarni) dan Ayahanda (Hartoyo), yang selalu mendukung, mendoakan anaknya yang sedang berjuang dalam studinya untuk dapat gelar sarjana serta adikku tercinta (Amalia Ichsani) yang selalu menyemangati dalam perjuangan ini.
3. Bapak Ahmad M Raf'ie Pratama, S.T., M.I.T, selaku pembimbing yang telah meluangkan waktu untuk memberikan pengarahan selama penyelesaian tugas akhir ini, terima kasih pak saya selalu berdoa untuk kesehatan dan kebahagiaan bapak.
4. Bapak Yudi Prayudi, S.Si., M.Kom., selaku Ketua Jurusan Teknik Informatika, Fakultas Teknologi Industri.

5. Bapak Raden Ratna Dewa, Bapak Ari Sujarwo dan Bapak Arwan Ahmad Khoiruddin yang senantiasa memberikan pencerahan serta semangat untuk terus berkarya.
6. Sahabat kalong Gardyas Bidari Adninda atas tawa, tangis, senyum, hina untuk mengingatkan agar penulis segera menyelesaikan tugas akhir ini.
7. Raisya yang senantiasa mendukung dan memberikan semangat untuk segera menyelesaikan tugas akhir serta mengajari arti fokus pada tujuan.
8. Yani yang telah mengajari bagaimana arti hidup dan saling menghargai.
9. Teman dan sahabat tim migrasi FTI UII (Martin, Yos, Andihary Talisman) atas semua bantuan, saran dan pemikiran-pemikiran selama ini.
10. Teman seperjuangan di SYNAPTIC UII (Agast, Laifa, Dita, Novi, Puput, Heru, Synaptic 2010, 2011 dan 2012).
11. Teman – teman lab SISJARKOM (Mas Dodo, Mas Hendra, Mas Hanafi, Mas Icong, Mas Irfan, Mas Valent, Mbak Nita, Mbak Kiki, Olip, Kikuk, Coco, Timur, Andot) terima kasih atas bantuan dan motivasinya.
12. Semua pihak yang telah membantu penulis dalam penyelesaian tugas akhir

Penulis menyadari dalam penyelesaian tugas akhir ini masih ada kekurangan, oleh karena itu penulis memohon kritik dan saran yang bisa membangun demi kesempurnaan tugas akhir ini.

Akhir kata semoga tugas akhir ini dapat memberikan manfaat bagi kita semua.

Yogyakarta, 5 Juli 2012

Penulis

SARI

Private cloud storage adalah sebuah servis yang dikembangkan dari *Infrastructure as a Service* (IaaS) yang menyediakan service untuk penyimpanan data agar data dapat diakses oleh user dari mana saja. Tipe penyimpanan dalam cloud storage terbagi menjadi tiga tipe yaitu *file storage*, *block storage* dan *object storage*. *Cloud storage* dapat diintegrasikan dengan layanan *collaborative tools* sehingga *user* dapat lebih mudah untuk berbagi sumber daya dan saling berkolaborasi. *Cloud storage* dapat menjadi sebuah solusi dalam penyimpanan data didalam jaringan lokal Jurusan Informatika.

Permasalahan yang dimiliki oleh Jurusan Informatika adalah tidak adanya penyimpanan data terpusat sehingga dosen serta mahasiswa kesulitan untuk berbagi sumber daya dan kegiatan transfer ilmu dalam hal unggah dan unduh data ketika melakukan kegiatan belajar mengajar secara *live coding* ataupun kegiatan pelatihan pengabdian masyarakat yang sering dilakukan oleh Jurusan Informatika.

Pada penelitian ini dilakukan pengujian fungsionalitas dan kinerja pada *private cloud storage* yang dapat diintegrasikan dengan layanan kolaborasi dan sistem terpusat. Pengujian dilakukan dengan membangun satu buah *private cloud storage* dengan tambahan layanan kolaborasi.

Dari hasil analisis menunjukkan bahwa *private cloud storage* ini memiliki fungsionalitas dan kinerja lebih baik daripada cloud storage semacam Dropbox, Owncloud, dan Swift yang tidak dibekali dengan layanan kolaborasi. Hal tersebut akan dibuktikan pada penelitian ini.

Kata kunci: *Private cloud storage*, kolaborasi, fungsionalitas.

TAKARIR

<i>application-development</i>	<i>pengembangan aplikasi</i>
<i>availability</i>	<i>ketersediaan sistem</i>
<i>client</i>	<i>klien</i>
<i>cloud computing</i>	<i>komputasi awan</i>
<i>cloud storage</i>	<i>penyimpanan data di dalam internet</i>
<i>collaboration software</i>	<i>perangkat lunak yang membantu kolaborasi</i>
<i>control panel</i>	<i>panel kontrol</i>
<i>mobility</i>	<i>mobilitas</i>
<i>multiplatform</i>	<i>sistem dapat berjalan disemua platform</i>
<i>pay-as-you-go</i>	<i>membayar apa yang digunakan</i>
<i>remote</i>	<i>pengendalian jarak jauh</i>
<i>resource</i>	<i>sumber daya</i>
<i>server</i>	<i>komputer yang menjadi penyedia layanan</i>
<i>sharing</i>	<i>berbagi sumber daya</i>
<i>standalone</i>	<i>sistem berjalan sendiri (tidak terhubung ke jaringan)</i>
<i>storage</i>	<i>tempat penyimpanan data</i>
<i>throughput</i>	<i>ukuran dari transfer bit yang diterima</i>

DAFTAR ISI

LEMBAR PENGESAHAN DOSEN PEMBIMBING	Error! Bookmark not defined.
LEMBAR PERNYATAAN KEASLIAN HASIL TUGAS AKHIR	iii
HALAMAN PERSEMBAHAN	v
MOTTO	vi
KATA PENGANTAR.....	vii
SARI.....	ix
TAKARIR.....	x
DAFTAR GAMBAR.....	xiii
DAFTAR TABEL	xv
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	3
1.6 Metodologi Penelitian.....	4
1.6.1 Studi Pustaka	4
1.6.2 Implementasi dan Konfigurasi Sistem.....	4
1.7 Sistematika Penulisan	5
BAB II LANDASAN TEORI.....	7
2.1 Cloud Computing	7
2.2 IaaS, SaaS, dan PaaS.....	9
2.3 Public Cloud, Private Cloud, dan Hybrid Cloud	11
2.4 Cloud Storage	13
2.5 Lightweight Directory Active Protocol (LDAP).....	15
2.6 Collaboration software.....	16
BAB III METODOLOGI	18

3.1	Analisis Masalah.....	18
3.2	Analisis Kebutuhan Sistem	21
3.2.1	Kebutuhan Antar Muka.....	21
3.2.2	Kebutuhan Data	21
3.3.3	Kebutuhan Fungsionalitas	21
3.3	Gambaran Umum Sistem	22
3.4	Desain Sistem	23
3.4.1	Alur Data dan Proses Aplikasi	24
3.4.2	Cara Kerja Layanan	26
3.4.3	Tingkat Keamanan.....	27
3.5	Analisis dan Persiapan Kebutuhan Sistem.....	28
3.5.1	Perangkat Keras Yang Dibutuhkan	28
3.5.2	Perangkat Lunak Yang Dibutuhkan.....	29
3.6	Pengembangan Sistem	31
3.7	Pengujian Sistem	32
3.7.1	Rencana pengujian sistem.	32
BAB IV HASIL DAN PEMBAHASAN.....		33
4.1	Implementasi Secara Umum	33
4.1.1	Pilihan Teknologi	34
4.2	Tahapan Implementasi Perangkat Lunak	38
4.3	Implementasi Hasil Perancangan.....	39
4.3.1	Instalasi dan konfigurasi Seafire server	39
4.3.2	Instalasi dan Konfigurasi OpenLDAP	43
4.4	Pengujian Hasil Implementasi.....	49
4.4.1	Pengujian fungsionalitas layanan Seafire server	49
4.4.2	Pengujian kinerja layanan Seafire server	53
BAB V KESIMPULAN DAN SARAN.....		58
5.1	Kesimpulan.....	58
5.2	Saran	58
DAFTAR PUSTAKA.....		60

DAFTAR GAMBAR

Gambar 2.1 Enam paradigma komputasi (diadaptasi dari Voas dan Zhang (2009))	8
Gambar 2.2 Taksonomi dalam komputasi awan	11
Gambar 2.3 <i>Control Panel</i> Amazon Web Service (AWS)	13
Gambar 2.4 <i>Control Panel</i> Dropbox	14
Gambar 2.5 <i>Control Panel</i> Owncloud	15
Gambar 2.6 Google Docs (Google Drive)	16
Gambar 3.1 Penyimpanan data konvensional	18
Gambar 3.2 Skema Storage as a Service	19
Gambar 3.3 Implementasi <i>cloud storage</i> as a Service	22
Gambar 3.4 Desain sistem	23
Gambar 3.5 Implementasi alur data LDAP	25
Gambar 3.6 Implementasi alur data <i>cloud storage</i>	26
Gambar 4.1 Lingkungan pembangunan sistem	33
Gambar 4.2 Konfigurasi “ <i>mounting</i> ” partisi pada fstab	40
Gambar 4.3 Hak akses pada /opt/objectdata	40
Gambar 4.4 Instalasi Seafile server	41
Gambar 4.5 Instalasi Seahub	42
Gambar 4.6 Admin <i>Control Panel</i>	43
Gambar 4.7 User <i>control panel</i>	43
Gambar 4.8 Instalasi OpenLDAP	44
Gambar 4.9 Konfigurasi domain dan IP pada LDAP server	44
Gambar 4.10 Instalasi phpLDAPadmin	45
Gambar 4.11 Isi konfigurasi /etc/phpldapadmin/config.php	45
Gambar 4.12 Konfigurasi <i>firewall</i>	45
Gambar 4.13 phpldapadmin <i>Control Panel</i>	46
Gambar 4.14 Konversi CSV ke LDIF	47
Gambar 4.15 Konversi csv menggunakan csv2ldif.exe	47

Gambar 4.16 LDIF Format untuk LDAP server	48
Gambar 4.17 Hasil export data yang dikonversi.....	48
Gambar 4.18 Konfigurasi tambahan dalam ccnet.conf	49
Gambar 4.19 User <i>Control Panel</i>	50
Gambar 4.20 Fitur sharing file dan dokumen untuk berkolaborasi	51
Gambar 4.21 Fitur contacts didalam Seafile.....	51
Gambar 4.22 Fitur group sebagai tempat diskusi <i>user</i>	51
Gambar 4.23 <i>Control Panel</i> administrator group	52
Gambar 4.24 Fitur diskusi didalam group	52
Gambar 4.25 Aplikasi Seafile client.....	52
Gambar 4.26 Fitur online collaboration tool	53
Gambar 4.27 Service yang berjalan pada <i>cloud storage server</i>	54

DAFTAR TABEL

Tabel 4.1 Teknologi pembuatan <i>private cloud storage</i>	34
Tabel 4.2 Perbandingan OpenLDAP dan <i>Active Directory</i>	35
Tabel 4.3 Perbandingan fitur XFS dengan Unix File System (UFS), Veritas File System (VxFS), dan Microsoft NT File System (NTFS).....	36
Tabel 4.4 Perbandingan <i>backend cloud storage</i>	38
Tabel 4.5 Rata-rata throughput dan peak unggah data.....	54
Tabel 4.6 Rata-rata throughput dan peak unduh data.....	55

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi informasi yang begitu pesat baik dalam segi perangkat keras (kecepatan prosesor, memori, kapasitas media penyimpanan) maupun dari sisi perangkat lunak. Salah satu pemanfaatan IT terbesar yaitu adalah penyimpanan data oleh masyarakat yang semakin hari semakin mengalami peningkatan. Gaya hidup masyarakat yang dinamis pun mempengaruhi kondisi bahwa data harus bisa diakses dari segala penjuru tempat. Hal ini bersinggungan dengan beralihnya metode penyimpanan konvensional (*standalone storage computing*) menjadi metode penyimpanan jaringan yang mempermudah ketersediaan akses data.

Keperluan akan media penyimpanan tidak bisa dikesampingkan. Dua paradigma baru dalam pemanfaatan media penyimpanan ternyata sangat membantu perusahaan dalam menginvestasikan kebutuhan perangkat penunjang bisnisnya. Yang pertama perusahaan mencari cara manajemen lonjakan pengaksesan data yang efisien dan bagaimana data bisa diakses oleh pihak enterprise. Kedua, bertambahnya kompleksitas pengelolaan media penyimpanan dalam jumlah besar dan makin cepatnya lalu lintas data mendorong nilai bisnis yang lebih dalam segi pengaplikasian perangkat lunak dan *service*. (Jon, 2006).

Begitu juga dengan wilayah akademik Jurusan Informatika UII yang memiliki lima jurusan strata S-1 dan dua jurusan S-2 beserta keseluruhan sub divisi jurusan tersebut. Dengan banyaknya sub divisi yang berada di lingkungan Jurusan Informatika UII, kebutuhan media penyimpanan publik yang dapat diakses dari mana saja sangat diperlukan. Dalam prakteknya beberapa kasus sering dijumpai sulitnya dalam pengaksesan data dikarenakan manajemen penyimpanan konvensional yang kurang baik sehingga menyebabkan kegagalan dalam pengaksesan data. Banyak kasus kegagalan dalam pengaksesan data dikarenakan media penyimpanan mengalami error secara fisik maupun error secara

teknis membuat institusi akademik Jurusan Informatika UII mencari solusi permasalahan ini.

cloud storage menjadi salah satu layanan yang baru dalam sebuah lingkungan bisnis yang ada saat ini. Sebagai layanan baru, *cloud storage* diharapkan mampu memberikan layanan seoptimal mungkin, sekalipun dalam keadaan terjadinya suatu bencana ataupun kegagalan sumberdaya, sehingga bisnis dalam perusahaan tersebut tetap bertahan dan keuntungan bagi perusahaan/instansi akan terus mengalir.

Disini *private cloud storage* akan mengambil peran dalam implementasinya di lingkungan Jurusan Informatika UII. *Private cloud storage* menawarkan sebuah penyimpanan data yang mendukung aspek *scalability* atau kemampuan *server* dapat ditingkatkan dari sisi performa, fungsi dan penambahan jumlah *user*, *flexibility* atau kemudahan dalam penggunaan *server cloud storage*, *availability* atau keberadaan *server* yang dapat menangani transaksi data selama 24 jam dan *backup* atau kemampuan *server* dalam menangani kegagalan pengaksesan data/sistem.

Komponen penyusun *private cloud storage* ini tidak hanya terpaku pada infrastruktur *private cloud storage* melainkan didukung dengan aspek yang lain yaitu keamanan sambungan data serta optimasi aplikasi. Beberapa aspek sudah dapat dipenuhi dikarenakan keberadaan infrastruktur jaringan lokal Jurusan Informatika UII sudah terbentuk. Hal yang dilakukan selanjutnya adalah bagaimana optimalisasi layanan aplikasi dan jaminan sambungan data yang aman.

1.2 Rumusan Masalah

Berangkat dari latar belakang penelitian yang telah dijabarkan diatas, memunculkan sebuah perumusan masalah dari penelitian ini sebagai berikut :

- a. Bagaimana membangun *cloud storage* sebagai sarana penyimpanan data di lingkungan Jurusan Informatika yang memiliki *transfer time* dan *average throughput* yang baik.

- b. Bagaimana sistem dapat memberikan layanan manajemen data user secara terpusat.

1.3 Batasan Masalah

Penelitian tugas akhir ini mempunyai batasan mengenai pengimplementasian proyek ini adalah sebagai berikut

- a. Pengembangan dan implementasi sistem hanya terdapat dalam jaringan lokal Jurusan Informatika UII.
- b. Melakukan analisis performa mengunduh dan mengunggah data dengan besar data 500MB dengan 25 komputer pada satu waktu.

1.4 Tujuan Penelitian

Tujuan yang akan dicapai dalam perancangan proyek penelitian ini adalah sebagai berikut:

- a. Membangun sebuah *cloud storage* didalam jaringan lokal Jurusan Informatika dengan fitur kolaborasi dan otentikasi user secara terpusat.
- b. Mengetahui performa rata-rata *throughput* dan *transfer time* saat unggah dan unduh data pada *cloud storage*.
- c. Diperolehnya layanan *cloud storage* di jaringan lokal Jurusan Informatika, terkait kebutuhan dosen, mahasiswa dan staff dalam penyimpanan data di lingkungan Jurusan Informatika.
- d. Mendapatkan data yang valid berdasarkan hasil uji statistik.

1.5 Manfaat Penelitian

Manfaat yang bisa diperoleh dari terlaksananya penelitian tugas akhir ini diantaranya:

- a. Diperolehnya layanan otentikasi terpusat untuk membantu dalam melakukan manajemen user dari *cloud storage*.

- b. *User* yang berupa dosen, mahasiswa dan staf dapat menyimpan data secara *online* di lingkungan lokal jaringan Jurusan Informatika serta dapat berbagi sumber daya kepada *user* lain.
- c. Memberikan kemudahan implementasi *single account* pada pengembangan jaringan jurusan informatika kedepan.

1.6 Metodologi Penelitian

Metode penelitian berfungsi sebagai penggambaran bagaimana langkah atau strategi yang dilakukan dalam penelitian. Dalam penelitian tugas akhir ini metodologi yang digunakan terbagi menjadi dua tahap, yaitu:

1.6.1 Studi Pustaka

Studi pustaka digunakan untuk menggali informasi yang terkait dengan penelitian, yaitu melalui buku-buku dan sumber-sumber yang ada di internet.

1.6.2 Implementasi dan Konfigurasi Sistem

Penelitian ini menggunakan *waterfall* model yang mencakup lima tahapan kegiatan, yaitu:

a. Analisis.

Pada tahap ini dilakukan identifikasi kondisi yang ada, definisi serta cakupan sistem yang akan dikembangkan, permasalahan yang muncul, kondisi infrastruktur jaringan dan peralatan TI, sumber data, serta potensi sumber daya manusia yang tersedia pada masing-masing sumber data.

b. Perencanaan.

Setelah proses analisis selesai dilakukan perencana pembangunan infrastruktur dengan memperhatikan hasil pada tahap analisis, baik mengenai potensi infrastruktur peralatan TI maupun jaringan pendukung sistem serta sumber daya yang terlibat di dalamnya.

c. Desain.

Proses desain sistem *private cloud storage* yang akan dikembangkan meliputi desain yang terstruktur, konfigurasi perangkat keras dan lunak, evaluasi seluruh subsistem jaringan, dan implementasi desain.

d. Implementasi.

Desain yang telah dikembangkan kemudian diimplementasikan, dimulai dari penyusunan rencana secara detail, pengadaan peralatan yang kurang, penyiapan struktur pangkalan data, dan penetapan strategi migrasi ke sistem baru.

e. Pemeliharaan.

Kegiatan ini dilaksanakan setelah sistem mulai diuji coba dan migrasi ke sistem *private cloud storage* telah dilakukan. Pada tahapan ini dilakukan pula evaluasi terhadap sistem, yang meliputi perangkat keras, perangkat lunak, jaringan telekomunikasi, kecepatan akses/transfer data dan download, arsitektur jaringan, konfigurasi jaringan, dan sebagainya.

1.7 Sistematika Penulisan

Pembuatan laporan penelitian ini mempunyai struktur dan kerangka penulisan yang dijelaskan seperti dibawah ini:

BAB I Pendahuluan

Membahas tentang latar belakang masalah, batasan masalah, rumusan masalah, tujuan penelitian serta manfaat dari penelitian dan metodologi penelitian yang diangkat menjadi materi penulisan laporan tugas akhir *Private cloud storage* dan Implementasinya di Jaringan Jurusan Informatika UII.

BAB II Landasan Teori

Membahas dasar-dasar teori yang digunakan dalam perancangan dan pembangunan sebuah *Private cloud storage*.

BAB III Metodologi

Memuat uraian tentang analisis masalah, gambaran umum sistem, analisis kebutuhan sistem yang mencakup kebutuhan perangkat keras, perangkat jaringan, dan perangkat lunak yang digunakan untuk membantu penyelesaian tugas akhir.

BAB IV Hasil dan Pembahasan

Memuat dokumentasi mulai dari tahap instalasi, konfigurasi dan pengujian terhadap aplikasi, yaitu dengan uji akses file pada *Private cloud storage*.

BAB V Penutup

Memuat kesimpulan-kesimpulan dari seluruh rangkaian proses implementasi perangkat lunak, baik pada tahap analisis, perancangan, implementasi, terutama pada analisis kinerja aplikasi. Bab ini juga membahas saran yang dapat digunakan oleh pihak yang berkepentingan maupun untuk peneliti terhadap kekurangan serta keterbatasan dalam penelitian ini.

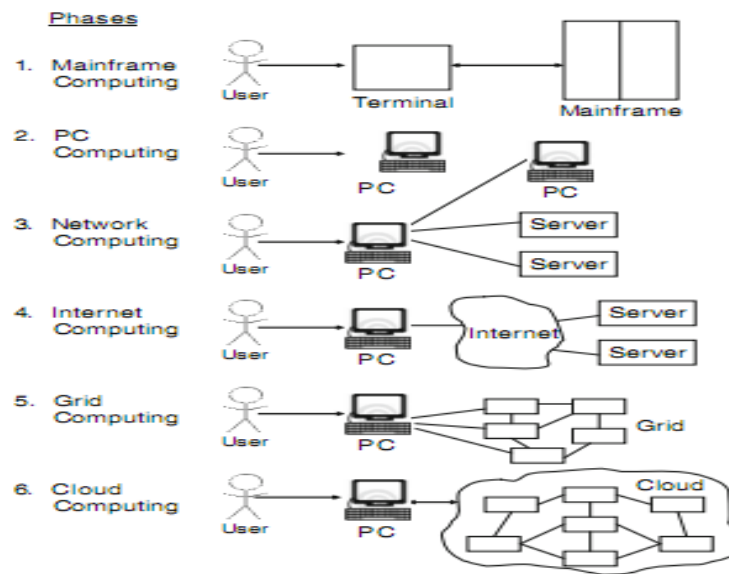
BAB II

LANDASAN TEORI

2.1 Cloud Computing

Cloud Computing merupakan paradigma baru dalam dunia komputasi. *Cloud computing* adalah teknologi informasi yang mengirimkan sumberdaya dan kemampuan sebagai servis melalui internet dimana data dikirimkan secara dinamis. *Cloud computing* juga disebut sebagai gaya komputasi yang dinamis, terukur dan sumber dayanya sering disediakan secara virtual melalui internet. Hal ini biasanya menggabungkan Infrastruktur sebagai layanan (IaaS), Platform sebagai layanan (PaaS), dan Perangkat lunak sebagai layanan (SaaS).

Dalam pengaplikasiannya *cloud computing* ini merupakan kumpulan dari beberapa *computing resource* yang terintergrasi sehingga menjadikannya sebagai layanan yang memiliki skalabilitas tinggi. Dengan teknologi *cloud computing*, *user* dapat menggunakan berbagai macam perangkat seperti PC, laptop, smartphones dan perangkat lainnya untuk melakukan akses terhadap perangkat lunak, penyimpanan data, dan platform untuk *application-development* yang ada di internet, melalui layanan yang diberikan oleh provider *cloud computing*. Keuntungan dari penggunaan *cloud computing* ini sendiri adalah perusahaan dapat melakukan penghematan biaya untuk pengadaan *resource* untuk komputasi, kemudian ketersediaan sistem dalam memberikan layanan akan terjamin dan kemudahan dalam skalabilitas.



Gambar 2.1 Enam paradigma komputasi (diadaptasi dari Voas dan Zhang (2009))

Skema diatas diadaptasi dari Voas dan Zhang (2009) , menampilkan enam fase dalam paradigma komputasi. Dari penggunaan dummy terminal kemudian ke PC hingga penggunaan *cloud computing*. Pada fase pertama, orang menggunakan mainframe dengan sumber daya komputasi besar yang kemudian sumber daya tersebut digunakan secara bersama-sama menggunakan dummy terminal untuk melakukan komputasi. Fase kedua adalah fase dimana penggunaan stand-alone PC sudah cukup untuk memenuhi kebutuhan *user* dalam melakukan komputasi. Fase ketiga adalah ketika PC, laptop dan *server* kemudian terkoneksi secara bersama-sama dalam sebuah jaringan lokal untuk berbagi sumber daya sehingga dapat meningkatkan performa komputasinya. Kemudian fase kelima adalah grid computing, grid computing menyediakan sumberdaya komputasi besar dan penyimpanan data yang dapat digunakan secara bersama-sama didalam komputasi sistem tersebar. Kemudian yang terakhir adalah fase keenam yaitu *cloud computing*, *cloud computing* menyediakan sumberdaya yang dapat digunakan bersama-sama melalui internet dengan skalabilitas tinggi dan dengan cara yang mudah.

Manfaat menggunakan *cloud computing* sangat menguntungkan jika ditinjau dari sisi finansial dan sisi efisiensi meliputi :

- Efisiensi

Dengan adanya *cloud computing* ini *end user* tidak memerlukan lagi masalah infrastruktur karena semua infrastruktur telah disediakan oleh provider. Dengan mekanisme ini pengelolaan waktu untuk *development* menjadi lebih banyak.

- *Mobility*

Semua *resource* yang tersimpan di dalam komputasi awan akan dapat diakses dimanapun selama *user* terhubung dengan internet.

- Pembiayaan sesuai dengan pemakaian

Dengan mekanisme *cloud computing* ini, pembiayaan adalah sesuai dengan apa yang *user* gunakan. Apa yang *user* pakai itulah yang *user* bayar. Hal ini yang menjadi salah satu pembeda antara traditional hosting dan mekanisme *cloud computing*.

- Meningkatkan *Availability*

Sistem *Cloud* telah di desain dengan kemampuan *high availability*. Dengan adanya fitur ini data atau aplikasi yang anda simpan di *cloud* akan di replika ke beberapa *server* sehingga data anda tetap aman karena jika suatu waktu *server* mengalami *crash*, maka *server* lain akan menggantikan peranan dari satu *server* tersebut.

2.2 IaaS, SaaS, dan PaaS

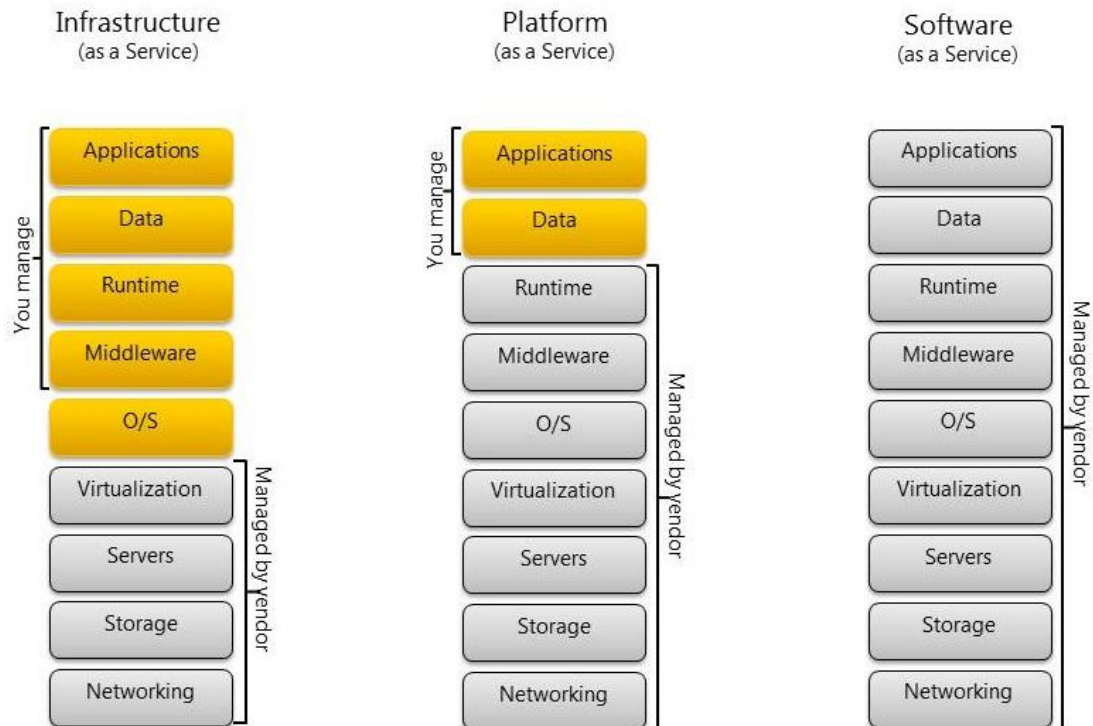
Dalam *cloud computing* terdapat beberapa layanan yang dapat dibagi menjadi beberapa bagian yang lebih spesifik. Beberapa *cloud provider* membagi layanan tersebut menjadi 3 bagian tersendiri yaitu *Infrastructure as a Service* (IaaS), *Platform as a Service* (SaaS) dan *Software as a Service* (PaaS).

IaaS atau *Infrastructure as a Service* adalah sebuah layanan yang menawarkan sumber daya komputasi sebagai layanannya. Pada layanan ini terdapat sebuah komputasi secara virtual dengan kemampuan pemrosesan data yang terjamin serta *bandwidth* yang tersendiri untuk penyimpanan data serta akses

internet. Sebagaimana konsep dalam *cloud computing* yang memberikan elastisitas, *user* bisa mengatur spesifikasi masing-masing komponen yang akan digunakan oleh *user* dan *user* dapat menambah atau mengurangi komponen dengan mudah. Misalnya Misalnya ketika *data storage* dari layanan yang *user* sewa sudah melewati batas kapasitas maksimum, *user* dapat menambahkan storage, atau CPU dengan mudah tanpa mengganggu proses yang sedang berjalan.

PaaS atau *Platform as a Service* mirip dengan IaaS, tetapi PaaS dilengkapi dengan sistem operasi dan membutuhkan layanan tertentu untuk menjalankan aplikasi tertentu. PaaS sendiri sebenarnya adalah IaaS yang dilengkapi dengan beberapa perangkat lunak untuk menjalankan aplikasi tertentu. Didalam PaaS, provider tidak hanya memberikan sistem operasi saja namun juga memberikan dukungan tertentu agar *user* mudah dalam mengembangkan sebuah sistem semisal database dan keperluan – keperluan lain yang berguna dalam pembuatan sebuah sistem.

SaaS atau *Software as a Service* adalah sebuah layanan yang memberikan kemudahan kepada *user* untuk menjalankan aplikasinya secara *remote* dari *cloud server* melalui internet. Dalam masalah biaya *user* akan dikenakan biaya tambahan ketika akan menggunakan fitur-fitur tambahan tertentu. Keuntungan bagi *user* ketika menggunakan SaaS adalah *user* tidak perlu membeli lisensi dari aplikasi yang berjalan di SaaS. Dalam perkembangannya nanti banyak aplikasi yang pada awalnya hanya dapat diakses melalui computer lokal akan dapat diakses melalui internet sebagai contoh adalah aplikasi semacam Office365, Salesforce dll.



Gambar 2.2 Taksonomi dalam komputasi awan

Perhatikan skema pada gambar 2.2. Dari skema diatas kita dapat dilihat bahwa setiap layanan memiliki perbedaan dalam sisi *user* dan provider. Dalam IaaS *user* melakukan pengelolaan dari sisi aplikasi hingga sistem operasi dan sisanya dikerjakan oleh *cloud* provider. Kemudian pada PaaS *user* melakukan pengelolaan pada aplikasi yang dijalankan dan data yang ada pada *cloud* tersebut, sedangkan SaaS *user* tidak melakukan pengelolaan terhadap sistem yang berjalan didalam *cloud*. Dalam sistem SaaS *user* akan memperoleh kemudahan karena pengelolaan dilakukan oleh *cloud* provider dan *user* dalam posisi tersebut adalah pengguna aplikasi atau end *user*.

2.3 Public Cloud, Private Cloud, dan Hybrid Cloud

Didalam *cloud computing* terdapat tiga tipe/model untuk *cloud computing* itu sendiri yaitu *public cloud*, *private cloud* dan *hybrid cloud*.

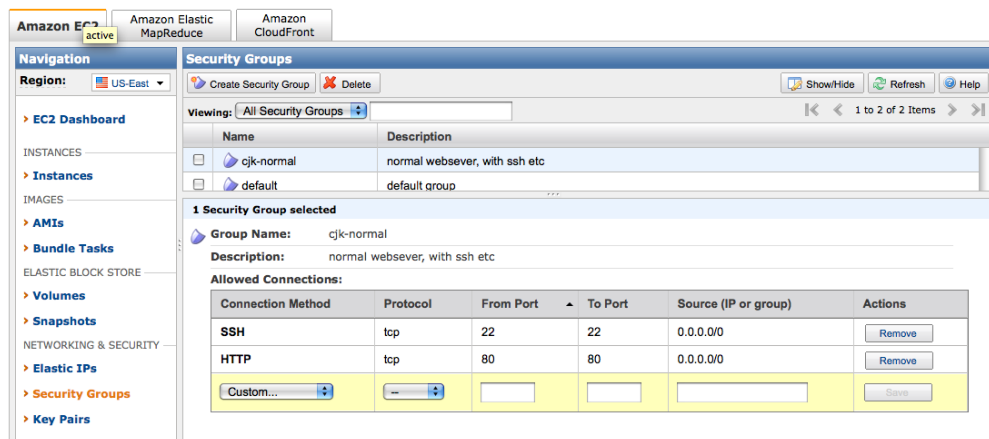
Merujuk pada kata *public*, *public cloud* atau eksternal *cloud* adalah bentuk paling umum dari Cloud Computing, dimana layanan yang dibuat tersedia untuk

umum dengan cara *pay-as-you-go*. User baik individu maupun perusahaan mengakses layanan yang disediakan oleh *cloud* provider melalui internet. Didalam *public cloud computing* sumber daya ditetapkan melalui internet dengan menggunakan aplikasi berbasis web ataupun web servis dari penyedia layanan. *Public cloud* berjalan melalui pihak ketiga, dan aplikasi yang berjalan didalamnya adalah aplikasi yang dimiliki oleh *user* yang berbeda dimana aplikasi tersebut menjadi satu didalam *cloud server*, sistem penyimpanan dan jaringan yang sama.

Private cloud atau internal *cloud* adalah sebuah *cloud computing* yang berjalan didalam sebuah jaringan lokal/*private* network. *Private cloud* dibangun untuk sebuah eksklusifitas penggunaan oleh satu klien, pada *private cloud computing* klien diberikan hak akses penuh pada kontrol data, keamanan data dan *quality of service*. *Private cloud computing* cepat dibangun dan dikelola oleh perusahaan yang memiliki divisi IT sendiri ataupun penyedia layanan *cloud* yang memberikan layanan *private cloud* didalam servisnya.

Hybrid cloud adalah sebuah lingkungan yang mengkombinasikan antara model *public cloud* dan *private cloud*. *Hybrid cloud* mengenalkan kompleksitas penentuan bagaimana cara untuk mendistribusikan aplikasi bisa berjalan pada *public cloud* dan *private cloud*.

Amazon Web Service merupakan salah satu contoh dari penerapan *cloud computing*. Amazon Web Service (AWS) adalah kumpulan layanan komputasi jarak jauh (disebut juga sebagai layanan web) yang bersama-sama membentuk sebuah platform komputasi awan, yang ditawarkan melalui internet oleh Amazon.com. Layanan yang paling utama dan terkenal adalah Amazon EC2 dan Amazon S3.



Gambar 2.3 Control Panel Amazon Web Service (AWS)

Layanan ini menyediakan kapasitas komputasi besar yang jauh lebih cepat dan lebih murah daripada membangun sebuah *server* farm secara fisik. Hal yang perlu diperhatikan dalam AWS ini adalah tersedianya *service* IaaS hingga SaaS, sehingga bisa dikatakan bahwa AWS memiliki *service* yang lengkap untuk melayani kebutuhan pengguna.

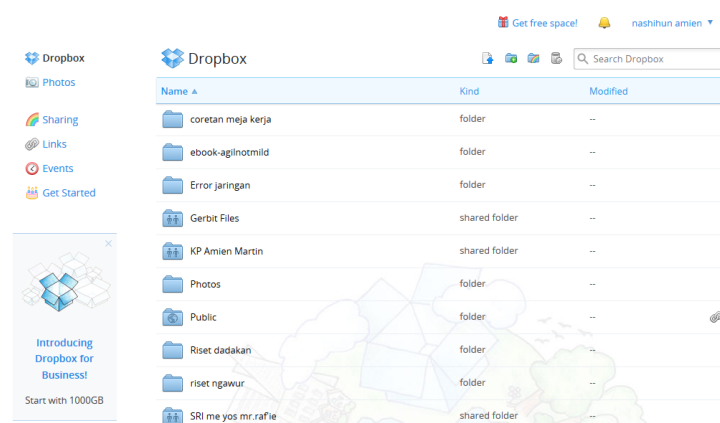
2.4 Cloud Storage

Cloud storage adalah model layanan dimana data dirawat, dikelola dan mendapat dukungan untuk pengaksesan jarak jauh serta dibuat untuk tersedia oleh pengguna melalui jaringan lokal maupun internet. Dalam perkembangan *cloud storage* solusi yang ditawarkan dibagi menjadi tiga bagian kategori yaitu *object storage*, *block storage* dan *file storage*.

Object-based storage device (OSD) atau bisa disebut juga *object storage* adalah sebuah perangkat penyimpanan pada komputer yang mirip dengan *disk storage* tetapi bekerja pada level yang lebih tinggi, dimana dengan penyimpanan model ini maka file akan diakses melalui interface HTTP yang biasanya dikerjakan oleh REST API dan semua akses klien dilakukan pada tingkat pengguna (*user*). Yang artinya sistem operasi tidak akan menyadari keberadaan dari sistem penyimpanan jarak jauh ini. Contoh *service* yang menyediakan *object storage* adalah Amazon S3, Rackspace Cloud Files dan Ceph Storage

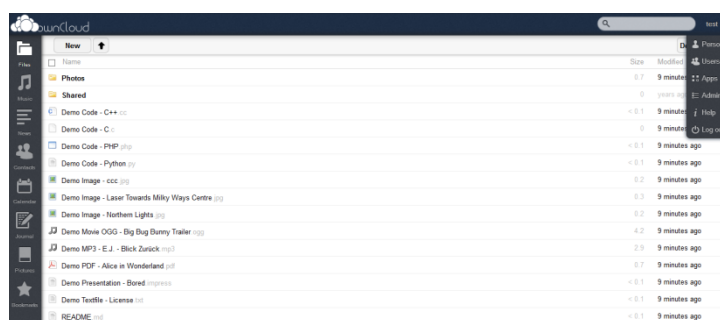
Block Storage adalah sebuah teknologi penyimpanan data dimana data dikelola menggunakan antar muka tingkat rendah dari sebuah komputer bus interface seperti SCSI ataupun ATA yang dapat diakses melalui sebuah jaringan. *Block storage* identik dengan SAN (*Storage Area Network*). Klien melakukan akses data melalui sistem operasi pada level perangkat yaitu *user* melakukan akses data menggunakan “*mounting*” perangkat yang berada dalam keadaan remote sebagaimana ketika *user* mengakses data dengan cara “*mounting*” secara lokal pada perangkat fisik. Contoh *service* yang menyediakan *block storage* sebagai layanannya adalah Amazon Elastic Block Store (EBS), Ceph RADOS Block Device (RBD), iSCSI.

Merujuk pada kata *file storage*, *file storage* adalah sebuah teknologi penyimpanan file dimana file diakses melalui protokol sistem file terdistribusi. File storage ini identik dengan NAS (*Network Attached Storage*). Klien melakukan akses data melalui sistem operasi pada tingkat sistem file yaitu pengguna melakukan akses data dengan cara melakukan “*mounting*” terhadap remote *file storage* dengan menggunakan perangkat lunak klien yang sesuai untuk dapat mengaksesnya. Dropbox dan Owncloud adalah salah satu contoh dari penerapan *file storage*, dimana klien hanya dapat mengakses data yang berada didalam akun yang dimilikinya melewati aplikasi yang telah disediakan oleh penyedia layanan tersebut dalam hal ini adalah dropbox dan owncloud.



Gambar 2.4 Control Panel Dropbox

Dropbox adalah sebuah layanan *cloud storage* yang menawarkan penyimpanan data berupa foto, dokumen, spreadsheet dan berbagai tipe file computer lainnya menggunakan PC, tablet ataupun *mobile device*. Dropbox bekerja dengan cara menyimpan salinan identik dari sebuah file pada computer klien serta Dropbox *cloud-based storage system* dan kemudian secara otomatis melakukan sinkronisasi data melalui sambungan data yang aman melewati internet.



Gambar 2.5 Control Panel Owncloud

Owncloud merupakan layanan *cloud storage* yang memiliki fitur hampir sama dengan Dropbox dimana kedua layanan ini juga memiliki fitur sinkronisasi data. Hal yang perlu diperhatikan adalah Owncloud memiliki fitur tambahan yang tidak dimiliki oleh Dropbox seperti kalender, kontak dan apps store. Di sisi lain Owncloud dapat dibuat menjadi sebuah *hybrid cloud computing* dimana akses data dapat dilayani secara *private* melewati jaringan lokal maupun secara publik melewati internet.

2.5 Lightweight Directory Active Protocol (LDAP)

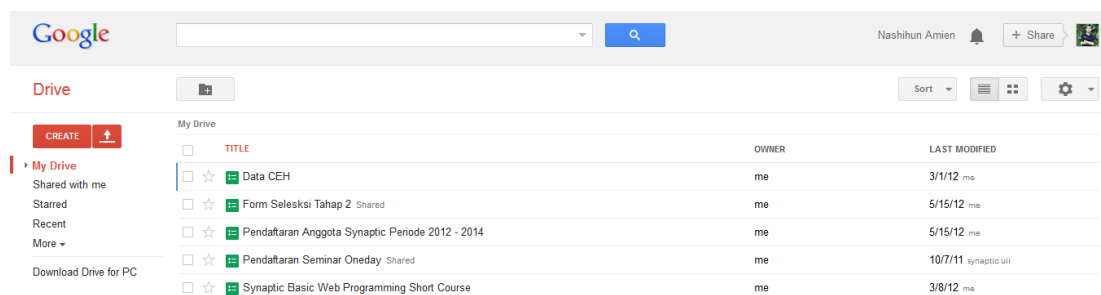
LDAP (*Lightweight Directory Access Protocol*) adalah protokol perangkat lunak untuk memungkinkan semua orang mencari *resource* organisasi, perorangan dan lainnya, seperti file atau printer di dalam jaringan baik di internet atau intranet. Protokol LDAP membentuk sebuah direktori yang berisi hirarki pohon yang memiliki cabang, mulai dari negara (countries), organisasi, departemen sampai dengan perorangan. Dengan menggunakan LDAP, seseorang dapat

mencari informasi mengenai orang lain tanpa mengetahui lokasi orang yang akan dicari itu. (Gerald, 2003)

2.6 Collaboration software

Collaborative software atau *groupware* adalah perangkat lunak yang dirancang untuk membantu orang yang terlibat dalam tugas bersama mencapai tujuan. Salah satu definisi awal dari *collaboration software* adalah sebuah kelompok yang sengaja mengerjakan sesuatu dengan menggunakan perangkat lunak tambahan yang mendukung mereka dalam mengerjakannya.

Tujuan dibuatnya *collaboration software* adalah untuk mengubah cara berbagi dokumen dan konten untuk sebuah kolaborasi dalam tim yang lebih efektif. Kolaborasi membutuhkan individu yang bekerja secara bersama secara terkoordinasi untuk mencapai tujuan bersama. *Collaboration software* membantu memfasilitasi tim berorientasi pada tindakan bekerja sama dengan jarak yang jauh dengan menyediakan alat bantuan komunikasi, kolaborasi dan proses pemecahan masalah. Selain itu, *collaboration software* dapat mendukung fungsi manajemen proyek seperti tugas, manajemen waktu dan kalender bersama.



Gambar 2.6 Google Docs (Google Drive)

Google Docs adalah sebuah *collaboration tools* yang dikembangkan Google untuk kebutuhan manajemen dokumen. Khususnya aplikasi office. Mulai dari pengolah kata, pengolah lembar kerja (*spreadsheet*) dan presentasi (*presentation*). Google Docs bukan hanya menyimpan saja, namun juga bisa digunakan untuk untuk mengolah, menyimpan, membuat, meng-edit

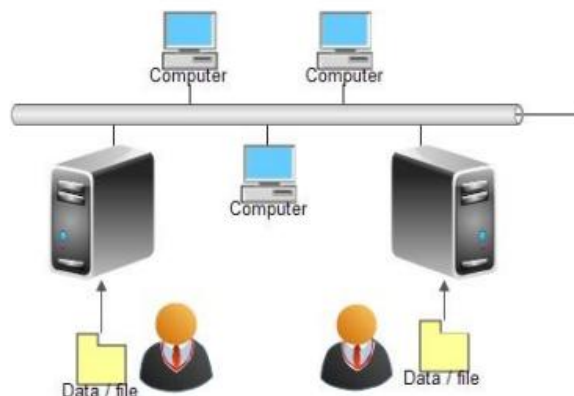
dokumen perkantoran seperti layaknya Microsoft Office pada Windows atau Open Office pada Linux dengan secara online. Hal yang perlu diperhatikan dari google docs adalah *user* dapat menggunakan google docs tanpa melakukan instalasi perangkat lunak, google docs berjalan melalui protokol http dan dapat diakses melalui akses internet.

BAB III METODOLOGI

3.1 Analisis Masalah

Standalone storage yaitu cara penyimpanan data pada komputer yang berdiri sendiri, tidak terhubung ke sebuah jaringan. Seiring tuntutan akan kebutuhan data yang besar dan cepat oleh *user*, penyimpanan data secara standalone ini memberikan akses yang terbatas akan kebutuhan *user* tersebut. Keterbatasan yang dimiliki membuat *user* tidak bisa mengakses data selain dengan *user* harus melakukan login di komputer kepemilikan sendiri tersebut.

Namun pada kenyataannya keberadaan infrastruktur jaringan yang sudah ada pada sebuah lingkungan kerja, institusi maupun perusahaan belum mampu memaksimalkan fungsionalitas teknologi yang dapat diterapkan dalam jaringan. Seperti halnya di lingkungan akademik Jurusan Informatika UII, infrastruktur jaringan lokal yang ada sementara ini hanya dimanfaatkan untuk akses internet. Dalam hal penyimpanan data pun masih dilakukan di komputer lokal meskipun komputer tersebut sudah terkoneksi kedalam jaringan. Penggambaran situasi ini bisa dilihat melalui skema pada Gambar 3.1.

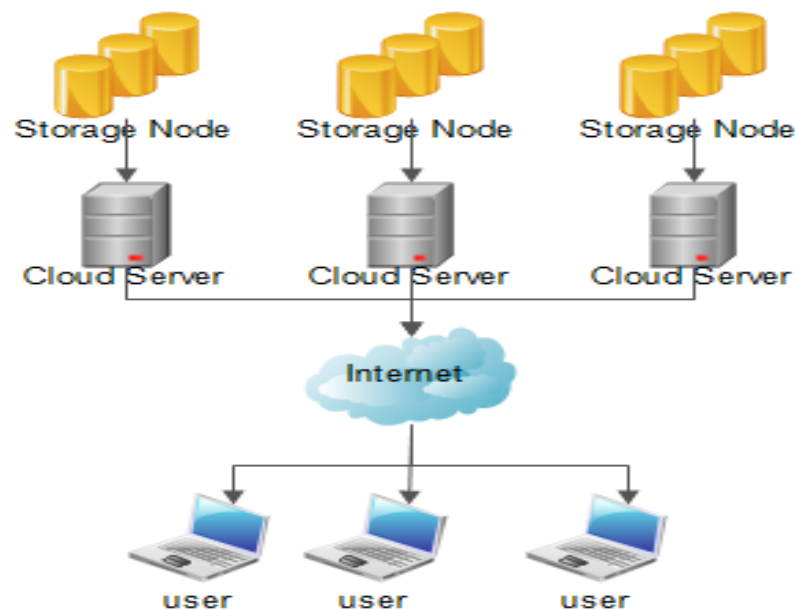


Gambar 3.1 Penyimpanan data konvensional

Praktek di lapangan sering dijumpai kasus kegagalan akses data dikarenakan manajemen penyimpanan konvensional yang kurang baik sehingga menyebabkan *data corrupt*. Banyak kasus kegagalan akses data dikarenakan *error* fisik maupun *error* teknis membuat institusi akademik Jurusan Informatika UII mencari solusi permasalahan ini. Kegagalan akses data karena rusaknya struktur data membuat keberlangsungan aktivitas pengguna terhambat, yang kemudian berakibat pada tidak maksimalnya kegiatan akademik Jurusan Informatika UII.

Cloud computing menjadi salah satu komponen dalam lingkungan bisnis yang ada saat ini. Sebagai layanan bisnis, *cloud computing* diharapkan mampu memberikan pelayanan seoptimal mungkin, sekalipun dalam keadaan terjadinya suatu bencana ataupun kegagalan sumber daya, sehingga bisnis dalam perusahaan tersebut tetap bertahan dan keuntungan bagi perusahaan/instansi akan terus mengalir.

Salah satu aspek utama dalam *cloud computing* ini adalah infrastruktur penyimpanan data. *Storage as a Service* (STaaS) menjadi terobosan teknologi penyimpanan skala besar yang mampu melayani akses data yang cepat. Skema *STaaS* dapat dilihat pada Gambar 3.2



Gambar 3.2 Skema Storage as a Service

Untuk menjawab permasalahan kegagalan akses data, fitur pendukung media penyimpanan *cloud computing* ini adalah backup dan sinkronisasi. Teknologi ini mendukung fitur toleransi kegagalan, artinya sistem mampu melakukan distribusi data ketika sebuah media penyimpanan mengalami kegagalan. Ketika media penyimpanan mengalami kegagalan, maka data hasil replikasi secara otomatis akan didistribusikan kepada media penyimpanan lain untuk memastikan bahwa data berada dalam keadaan aman. Untuk kepentingan *cloud storage* akan ditangani oleh *Seafile server* dimana semua data akan disimpan dan diakses melalui layanan ini.

Masalah selanjutnya yang muncul di dalam Jurusan Informatika UII adalah tidak adanya *collaboration tools* privat yang mendukung jalannya proses pembelajaran di lingkungan Jurusan Informatika UII. Hal ini membuat proses transfer ilmu antara mahasiswa dengan dosen maupun antar mahasiswa kurang dapat dimaksimalkan.

Solusi yang ditawarkan dari masalah kedua ini adalah membangun sebuah portal dengan layanan yang memudahkan pengguna dilingkungan Jurusan Informatika UII untuk berbagi sumber daya serta melakukan kolaborasi sehingga pekerjaan yang dikerjakan dapat mencapai hasil yang maksimal. Untuk kepentingan layanan kolaborasi dan berbagi sumberdaya akan ditangani oleh *Seafile Client* melalui protokol http.

Permasalahan selanjutnya penyimpanan jaringan yang merupakan lokasi yang dilalui oleh ribuan *user* dalam satu waktu menimbulkan tingginya resiko keamanan dan kerahasiaan data. Keberadaan portal yang berada di pusat jaringan publik menuntut kemudian dikembangkannya juga sistem keamanan akses data namun dengan tetap memperhatikan kemudahan akses tiap penggunanya.

Solusi yang ditawarkan dari permasalahan ketiga ini adalah membangun layanan otentikasi secara terpusat dan memberikan layanan dengan data terenkripsi sehingga data yang dikirim *user* sehingga *user* yang tidak berwenang tidak dapat membaca data disimpan dalam *cloud storage*. Untuk kepentingan

akses data akan dilakukan otentikasi melalui LDAP server, dimana LDAP server akan melakukan manajemen akun pengguna *cloud storage* tersebut.

3.2 Analisis Kebutuhan Sistem

3.2.1 Kebutuhan Antar Muka

Kebutuhan – kebutuhan antar muka untuk membangun *cloud storage* ini adalah sebagai berikut :

- 1) *Cloud storage* yang dibangun harus mempunyai tampilan yang familiar untuk pengguna.
- 2) Kebutuhan antar muka untuk kolaborasi dengan sesama *user* dengan menggunakan grup.
- 3) Panel kontrol user untuk melakukan *sharing* data dengan user lain dan manajemen konten yang dimiliki oleh user serta melakukan manajemen terhadap akun yang dimiliki oleh *user* tersebut

3.2.2 Kebutuhan Data

Data yang diolah oleh cloud storage ini adalah :

- 1) Data dan Objek yang diunggah user kedalam *cloud storage* dapat berupa gambar, file suara, dan objek digital lainnya.
- 2) Data dan file yang dikerjakan user dengan menggunakan *collaboration tools* yang ada didalam *cloud storage*.
- 3) Data *user* yang didaftarkan dengan menggunakan LDAP.

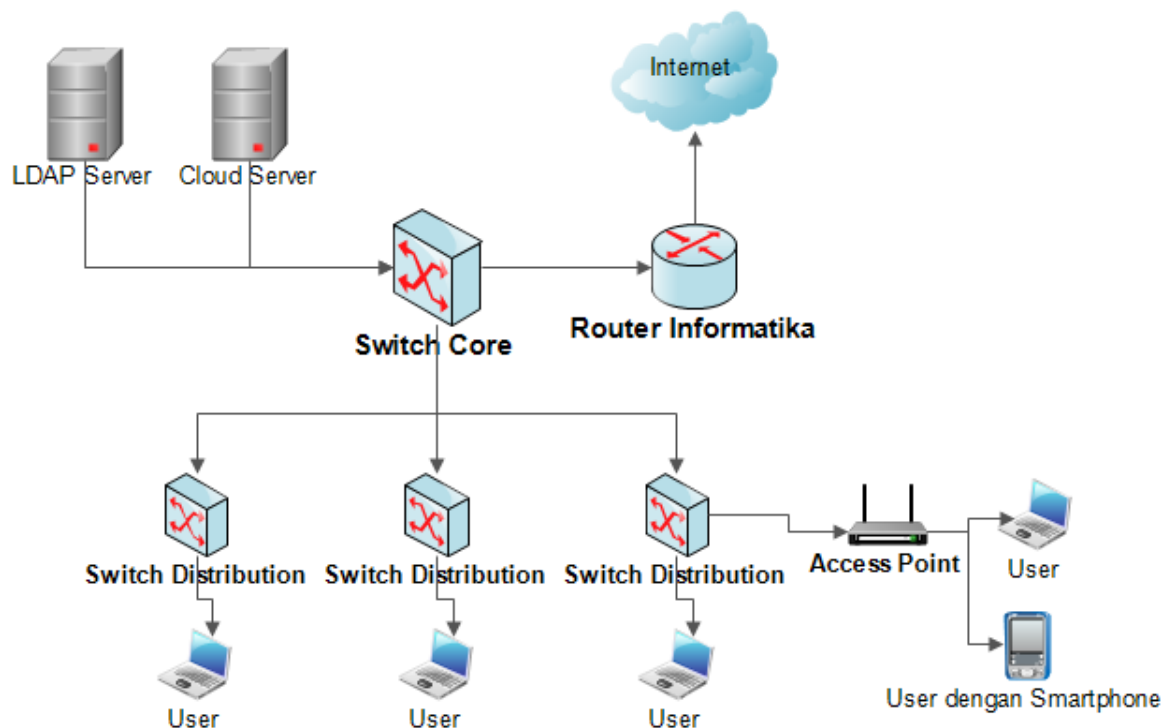
3.3.3 Kebutuhan Fungsionalitas

Fungsi – fungsi yang dimiliki oleh cloud storage ini adalah :

- 1) Mengelola otoritas *passcode*, yaitu data *passcode* yang dimasukkan oleh operator dengan benar, dan selanjutnya memilih menu untuk melakukan aktifitas selanjutnya.
- 2) Mengelola data dan objek yang berada didalam *cloud storage*.
- 3) Sinkronisasi data dengan *native app*.

- 4) *Online collaboration tool* sebagai sarana user berkolaborasi dengan user lainnya dalam melakukan pekerjaan ataupun untuk sekedar berbagi sumber daya.
- 5) Pengelolaan grup oleh user admin.

3.3 Gambaran Umum Sistem



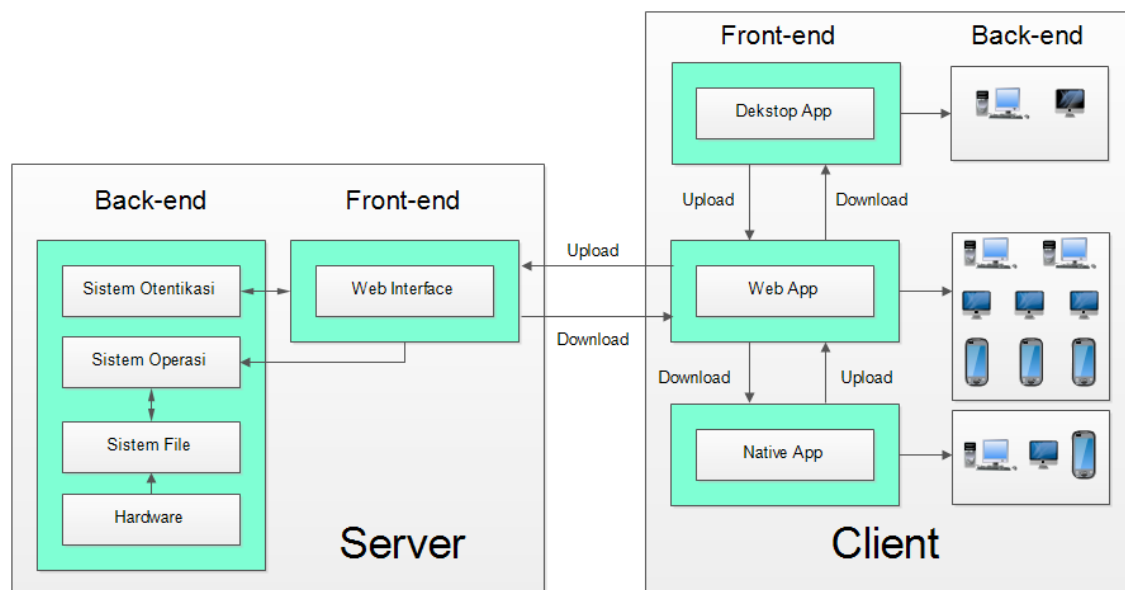
Gambar 3.3 Implementasi *cloud storage* as a Service

Cloud server sebagai media penyimpanan data di jaringan yang berkapasitas 1 TB dengan 1 TB sebagai tempat penyimpanan data *user*. *Server cloud computing* ini mempunyai 1 buah network interface card (NIC). 1 buah NIC ini berfungsi untuk interface yang terhubung dengan switch core, interface yang dapat diakses oleh *user*. Alamat IP interface ini disetting statik, disesuaikan dengan IP address yang dikenali oleh switch dan router Jurusan Informatika UII.

Aplikasi *web browser* di sisi *user* digunakan dalam mengakses data yang ada didalam *cloud storage* melalui protocol http.

3.4 Desain Sistem

Dari hasil perencanaan yang telah dipaparkan pada gambaran umum sistem, desain implementasi sistem dirancang menyesuaikan keadaan lingkungan objek penelitian (area Jurusan Informatika UII). Desain implementasi harus memperhatikan faktor kesederhanaan infrastruktur namun tetap menjaga keoptimalan sistem dan ketepatan penggunaan teknologi.



Gambar 3.4 Desain sistem

Sistem terbagi menjadi dua bagian yang pertama adalah bagian klien dan bagian selanjutnya adalah dari sisi *server*. Dari sisi klien kita dapat melihat pada Gambar 3.4 klien dapat mengakses sistem dengan menggunakan 3 metode yang pertama adalah menggunakan *web app* sebagai basis dari kontrol *cloud storage* yang akan dibuat kemudian user dapat menggunakan *desktop app* sebagai saran sinkronisasi data yang ada didalam *cloud* dan yang terakhir dengan menggunakan *native app* dimana user dapat melakukan manajemen data dan melakukan unggah ataupun unduh data melalui media *smart phone*.

Pemrosesan data dari *native app* maupun *desktop app* akan dilayani menggunakan protokol http oleh *server*. Dari sisi server data yang akan di unggah ataupun di unduh akan dilayani oleh *server* jika otentikasi *user* dan *passcode*

sesuai dengan yang telah didaftarkan di dalam sistem otentikasi. Didalam *server* sistem operasi berfungsi untuk melakukan manajemen perangkat keras dalam hal penyimpanan data serta. Sistem file sendiri mempunyai fungsi sebagai pendukung dari sistem operasi dalam menangani jumlah data yang masuk kedalam perangkat keras penyimpanan data.

3.4.1 Alur Data dan Proses Aplikasi

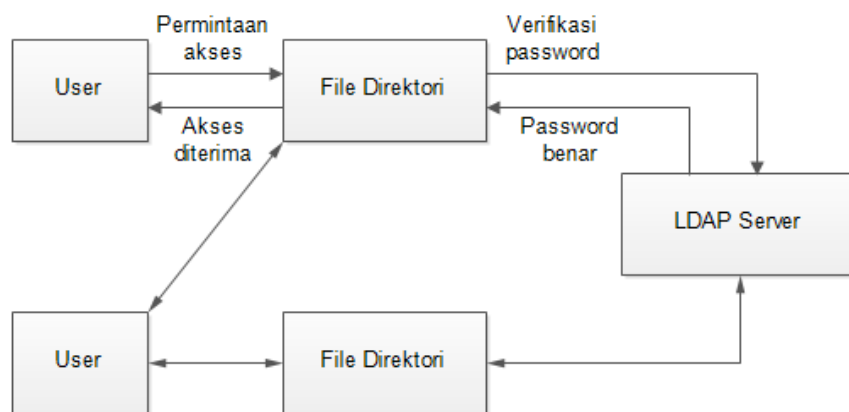
3.4.1.1 Sistem Otentikasi LDAP

Elemen dasar dari direktori aktif adalah *Active directory Object*. Sebuah *Active directory Object* ini dapat berupa sebuah akun pengguna, komputer yang tergabung ke dalam sebuah domain LDAP *Server*, printer, aplikasi, folder, atau sumber daya lainnya di dalam jaringan. Setiap objek memiliki atributnya masing-masing yang berupa properti yang umumnya bersifat unik (tergantung jenis objek tersebut). Sebagai contoh, atribut yang dapat dimiliki oleh objek akun pengguna dapat mencakup nama pertama, nama akhir, alamat e-mail, dan nomor telepon. Beberapa atribut lainnya memiliki nilai yang telah ditentukan oleh sistem, dan atribut lainnya dapat didefinisikan secara manual (atau bisa dikosongkan). Sementara itu, atribut yang dimiliki oleh objek printer dapat mencakup lokasi di mana printer tersebut berada, nomor inventaris aset, jenis printernya, dan lain sebagainya. *Active directory* juga memiliki peraturan-peraturan yang menata objek mana saja yang dapat disimpan di dalam direktori dan atribut mana saja yang dapat dimiliki oleh objek tersebut. Peraturan-peraturan tersebut, dinamakan juga dengan *Active directory Schema*.

Sebuah jenis objek khusus yang dapat disimpan di dalam *Active directory* adalah sesuatu yang disebut sebagai *Organizational Unit* (OU). OU adalah sebuah jenis objek *Active directory* yang dapat mengandung objek lainnya, seperti halnya sebuah akun pengguna, komputer, atau aplikasi atau mengandung objek OU lainnya. Dengan menggunakan OU, pada Administrator dapat mengatur *Active directory* secara hierarkis, yang disusun dengan menggunakan skema X.500 yang dibuat oleh International Telecommunication Union (ITU). Administrator juga

dapat menetapkan kontrol akses atau permissi pada setiap anak pohon dalam sebuah OU, agar hanya objek-objek tertentu saja yang dapat mengaksesnya.

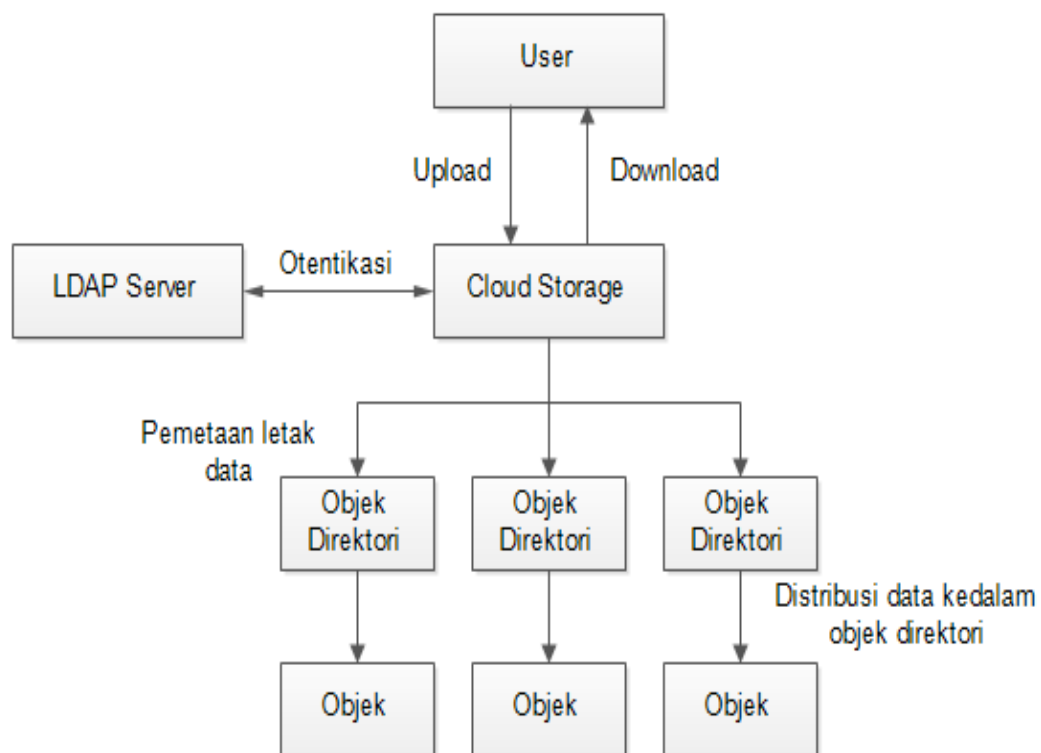
OU ditampung dalam sebuah entry, yang merupakan struktur dasar dari Active directory (pada kenyataannya, *Active directory* tidak akan berjalan tanpa adanya domain). Setiap objek di dalam *Active directory* harus termasuk ke dalam sebuah domain yang sama.



Gambar 3.5 Implementasi alur data LDAP

3.4.1.2 Alur Data Layanan *cloud storage*

Skema alur kerja pada Gambar 3.6 menjelaskan bahwa pengguna harus melakukan otentikasi sebelum mendapatkan akses kedalam sistem *cloud storage* tersebut.



Gambar 3.6 Implementasi alur data *cloud storage*

Proses otentikasi menyaring *user* berdasarkan informasi *user* yang ada didalam *LDAP server* menggunakan email dan password *user* yang sudah didaftarkan didalam *LDAP server*. Setiap akun memiliki akses yang berbeda kedalam sebuah halaman *Control Panel* yang dimiliki oleh masing-masing akun. Ketika *user* melakukan unggah data kedalam *server* maka *user* akan mengirimkan data kedalam *server* yang kemudian data tersebut akan ditempatkan kedalam sebuah objek direktori yang kemudian akan dipetakan kedalam database sistem. Begitu juga pada saat *user* melakukan pengunduhan data maka *user* akan melakukan otentikasi data kemudian *user* akan mengakses database *cloud storage* untuk memastikan dimana letak objek yang berada didalam objek direktori.

3.4.2 Cara Kerja Layanan

Elemen penting dalam *private cloud storage* ini adalah data dir atau blocks dan enkripsi data. Block dalam *private computing* ini berfungsi sebagai tempat dimana data *user* akan diletakkan. Ketika *user* mengunggah data kedalam *server*

maka data akan dienkripsi terlebih dahulu menggunakan password yang dimiliki *user* kemudian data akan di kirim kedalam *server* dan dialokasikan kedalam blok secara random untuk menghindari penggunaan data oleh pihak yang tidak memiliki hak akses kedalam file tersebut (lihat Gambar 3.6). Ketika data dikirimkan ke *server* maka *server* akan menyimpan informasi keberadaan file dan informasi *user* kedalam sebuah metadata didalam file basisdata.

Server cloud storage ini menggunakan password yang dimiliki *user* yang tersimpan didalam LDAP *server* untuk melakukan enkripsi / deskripsi semua isi file yang ada didalam library menggunakan algoritma AES-128. Token akan di-generate dari password dengan algoritma time consuming dan akan dikirim kedalam library. Ketika *user* melakukan pengecekan data, *server* akan melakukan pengecekan menggunakan token untuk memastikan bahwa password yang dimiliki *user* adalah benar. Waktu pengecekan ini membutuhkan 0.5 detik, hal ini didesain untuk mencegah serangan brute-force.

Ketika *user* melakukan sinkronisasi data dengan desktop client, password akan dikirimkan kedalam desktop *user* dan akan digunakan untuk melakukan enkripsi data sebelum *user* mengunggah data kedalam *server*.

3.4.3 Tingkat Keamanan

Masalah yang mungkin timbul pada sisi keamanan sistem ini adalah ketika pengguna mendapatkan akses ke pusat penyimpanan data jaringan, karena sangat beresiko pada keberadaan data-data yang bersifat rahasia dari beberapa *user*, maka dari itu sistem ini didesain menyembunyikan keberadaan komputer *server private cloud storage* dari akses langsung client, dengan menerapkan pengalamanan IP *private* pada *server*. Client mendapatkan akses data dari komputer *private cloud storage* melalui pintu gerbang layanan Web Server maupun layanan software pihak ketiga. Lihat skema gambaran sistem pada Gambar 3.3.

Pada sisi layanan *private cloud storage*, keamanan akses dilakukan dengan menerapkan otentikasi, yaitu sebuah prosedur pengecekan *username* dan

password melalui LDAP *server* sebelum client mendapatkan akses ke sebuah sistem. Prosedur keamanan lebih ditingkatkan dengan penambahan konfigurasi firewall baik di sisi *server* maupun di router pusat lalu lintas data di lingkup Jurusan Informatika UII.

3.5 Analisis dan Persiapan Kebutuhan Sistem

3.5.1 Perangkat Keras Yang Dibutuhkan

Berdasarkan Gambar 3.3 dan gambaran umum sistem, sudah jelas terlihat bahwa perangkat-perangkat keras yang dibutuhkan untuk menunjang implementasi dari semua fitur yang ada pada sistem ini adalah sebagai berikut:

3.5.1.1 Komputer Server Private Cloud Computing

Komputer ini merupakan computer yang digunakan sebagai pusat penyimpanan data pada jaringan. Kebutuhan minimal untuk membangun sebuah *server private cloud storage* yaitu:

- Processor 64 bit dual quad core
- Memory 4 GB RAM
- Network 1 GB Network Interface Card (NIC)
- 30 GB ruang hard disk kosong untuk instalasi OS Ubuntu beserta aplikasi tambahan yang dibutuhkan
- 1 TB ruang harddisk kosong sebanyak satu buah untuk media penyimpanan obyek

3.5.1.2 Komputer sebagai *Client* 1

Komputer yang digunakan untuk pengujian sistem menggunakan sistem operasi berplatform windows 8

- Processor 32 bit Core 2 Duo 2.20GHz
- Memory 2 GB RAM
- 250GB Ruang harddisk
- Sistem Operasi menggunakan Windows 8

3.5.1.3 Smartphone sebagai Client 2

Smartphone yang digunakan untuk pengujian sistem menggunakan sistem berplatform android.

- Processor Dual-core 1 GHz Cortex-A9
- Memory 1 GB DDR2 RAM
- 16 GB Ruang Harddisk
- Sistem Operasi yang digunakan Android OS v4.0 (Ice Cream Sandwich)

3.5.2 Perangkat Lunak Yang Dibutuhkan

3.5.2.1 Seafile Server

Seafile adalah sebuah aplikasi *client-server* untuk sinkronisasi file dan kolaborasi dengan lisensi Open Source. Seafile adalah sebuah alternatif dari owncloud atau dropbox.

Para pengembang menjelaskan bahwa Seafile memiliki fitur yang disebut kolaborasi online untuk kerjasama tim. Fitur ini tidak tersedia di Dropbox ataupun Owncloud. Fitur dari Seafile sebagian besar diakses melaluo web browser dan terlihat sangat mirip dengan GIT namun memiliki lebih banyak keuntungan bila dibandingkan dengan GIT. Dengan menggunakan Seafile *user* dapat dengan mudah mengunggah dan membuat file/folder baru, membuat perpustakaan, melihat perubahan dalam file/folder, membuat atau bergabung dalam kelompok, menambah kontak, pemberitahuan kelompok dan lain sebagainya.

3.5.2.2 Seafile Client

Seafile *client* adalah sebuah *native apps* yang dapat berjalan pada Linux, Windows, Mac, iOS dan Android. *Client* ini terdiri dari tray icon yang digunakan untuk menampilkan status sinkronisasi dan memungkinkan kita membuka klien didalam browser.

3.5.2.3 OpenLDAP Server

OpenLDAP merupakan salah satu *software* yang menerapkan protokol LDAP (Lightweight Directory Access Protocol) yang bersifat OpenSource dan tersedia diseluruh sistem operasi Linux. OpenLDAP memiliki bentuk struktur yang berhirarki (sistem pohon seperti pada file sistem linux), bukannya berformat kolom dan baris, seperti halnya database normal, sehingga memudahkan untuk memasukkan sejumlah besar detail yang mirip dalam bentuk yang terorganisir.

Di dalam OpenLDAP terdapat 2 *service* utama yaitu : slapd dan slurp. Slapd merupakan OpenLDAP daemon yang melayani permintaan dari klien, query dan berkomunikasi dengan backend database. Sedangkan slurp merupakan replication daemon yang berfungsi melayani replikasi data agar terus terjadi sinkronisasi data antara klien dan server.

Dengan penggunaan OpenLDAP dalam suatu sistem, maka akan memudahkan sistem tersebut dalam melakukan manajemen pengguna, karena data pengguna terpusat pada satu sistem. Atau biasa disebut sebagai Single Sign On.

3.5.2.4 phpLDAPadmin

phpLDAPadmin (PLA) adalah sebuah LDAP client berbasis web. PLA menyediakan kemudahan, dapat diakses dari mana saja dan administrasi multi bahasa untuk *server* LDAP.

PLA menampilkan data dengan hierarki pohon dan fitur pencarian yang membuat LDAP lebih mudah untuk dioperasikan. PLA dapat berjalan pada segala jenis platform mengingat PLA berbasis web sehingga PLA dapat dikelola dari lokasi manapun.

Beberapa fitur yang ditawarkan oleh PLA adalah sebagai berikut:

- LDAP tree browser
- Copy dan delete LDAP entries
- LDAP searches
- LDIF and DSML exportLDIF import

- Manage *user* password hashes (supports sha, crypt, md5, blowfish, md5crypt)
- Dan lain-lain

3.5.2.5 CSV2LDIF

Merujuk pada nama dari perangkat lunak tersebut CSV2LDIF adalah sebuah tools yang berfungsi untuk konversi data dari file csv kedalam ldif yaitu format yang digunakan oleh LDAP.

3.5.2.6 NTOP

Ntop adalah aplikasi yang digunakan untuk melakukan monitoring pada jaringan yang ada. Ntop menampilkan host yang sedang menggunakan jaringan dan memberikan lapotan informasi tentang trafik IP (internet protocol) dan Fibre channel (FC) yang dihasilkan tiap host.

3.5.2.7 Mozilla

Mozilla Firefox Merupakan software penjelajah web gratis yang multi platform. Pengembangan aplikasi ini bertujuan untuk mengembangkan sebuah peramban web yang kecil, cepat, sederhana. Dukungan multi platform ini dibuktikan dengan Firefox dapat dijalankan pada berbagai macam sistem operasi seperti Microsoft Windows, Linux, Mac OS X, dan FreeBSD.

3.6 Pengembangan Sistem

Model yang diusulkan diatas akan digunakan dalam tahap berikutnya, pengembangan sistem dan implementasi. Kedua langkah akan dilakukan pada *server* yang terletak didalam data center Jurusan Informatika Fakultas Teknologi Industri yang terdiri dari beberapa *server*. Spesifikasi detail dari *server* ini akan dibahas lebih lanjut pada sub bagian bab berikutnya.

Perkembangan sistem yang didasarkan pada model yang diusulkan pada proyek ini akan disesuaikan dengan lingkungan implementasi sendiri. Oleh karena

itu, beberapa alat khusus yang digunakan pada sistem ini hanya dapat sesuai dengan lingkungan Jurusan Informatika UII atau serupa dengan yang berada di dalam lingkungan Jurusan Informatika UII. Namun dengan sistem yang masih umum, yang berarti dapat disesuaikan dengan setiap jenis sistem jaringan karena menggunakan protokol LDAP dan HTTP yang merupakan protokol standar dalam manajemen *user* dan penyajian data dalam bentuk web.

3.7 Pengujian Sistem

Tahap terakhir pada desain sistem ini adalah untuk melakukan evaluasi diri dari proyek yang berdasarkan pada tahap implementasi. Evaluasi ini akan dilakukan dengan menentukan seberapa jauh sistem ini dapat melayani *user* untuk dapat berbagi sumber daya dan saling berkolaborasi. Evaluasi ini juga akan melibatkan beberapa pengukuran kinerja alat. Dari evaluasi ini, keterbatasan hasil yang dicapai oleh proyek ini dapat diidentifikasi. Dan hasil dari pengerjaan proyek ini dapat berguna untuk peluang penelitian selanjutnya.

3.7.1 Rencana pengujian sistem.

Setelah melakukan instalasi dan konfigurasi maka langkah selanjutnya adalah melakukan pengujian. Pengujian dilakukan dalam beberapa kondisi, yaitu :

1. Komputer klien menggunakan Windows 8 menggunakan aplikasi Mozilla untuk terkoneksi kedalam Seafile *control panel* dan Seafile client untuk melakukan sinkronisasi data.
2. Pengujian kinerja dilakukan dengan melakukan kegiatan unduh dan unggah data dengan menggunakan *dummy data* sebesar 100MB. sebanyak 2 kali percobaan dengan 25 komputer secara bersamaan. Dengan pengambilan data setiap 3 menit.
3. Pengujian fungsionalitas dilakukan dengan melakukan kegiatan *user-model oriented*, yaitu mencoba seluruh fitur yang diberikan oleh *cloud storage* dengan tambahan fitur kolaborasi.

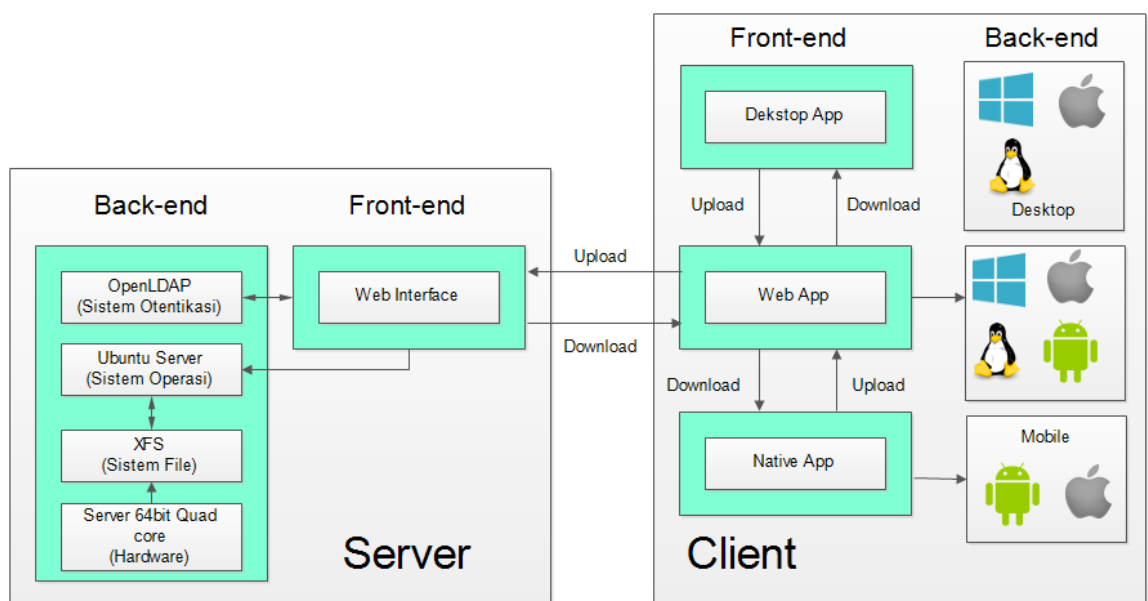
BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi Secara Umum

Implementasi sistem merupakan tahap dimana sistem mampu diaplikasikan dalam keadaan yang sesungguhnya. Dari implementasi ini akan diketahui apakah sistem yang dibuat dapat berjalan dengan baik atau tidak. Serta apakah sistem menghasilkan *output* yang sesuai dengan perancangan yang telah dibuat.

Implementasi sistem merupakan tahap dimana sistem mampu diaplikasikan dalam keadaan yang sesungguhnya. Dari implementasi ini akan diketahui apakah sistem yang dibuat dapat berjalan dengan baik atau tidak. Serta apakah sistem menghasilkan *output* yang sesuai dengan perancangan yang telah dibuat.



Gambar 4.1 Lingkungan pembangunan sistem

Dari Gambar 4.1 dari sisi *server* menggunakan beberapa teknologi untuk menangani sistem otentikasi, sistem operasi dan sistem file sehingga sistem dapat berjalan dengan semestinya. Pada tabel 4.1 dapat dilihat

Tabel 4.1 Teknologi pembuatan *private cloud storage*

No .	Nama Modul	Teknologi
1.	Sistem Operasi	Ubuntu Server 13.04 Linux
2.	Sistem Otentikasi	OpenLDAP Server + phpLDAPadmin
3.	Sistem File	Sistem File XFS
4.	Cloud Storage	Seafile server

4.1.1 Pilihan Teknologi

Teknologi yang digunakan untuk pembangunan *cloud storage* ini mencakup sistem operasi untuk manajemen perangkat keras penyimpanan data dan aplikasi yang dibutuhkan untuk menjalankan sistem, sistem otentikasi digunakan untuk manajemen data *user*, *passcode* serta *mail* untuk melakukan otentikasi kedalam *server cloud storage*, sistem file yang berfungsi untuk menangani data beserta meta data yang ada didalam data yang disimpan didalam *cloud storage*.

4.1.1.1 Linux Ubuntu Server 13.04

Ubuntu *server* adalah ubuntu yang didesain untuk diinstal kedalam *server*. Ubuntu sendiri adalah sebuah sistem operasi turunan dari sistem operasi Debian. Di dalam ubuntu *server* user bekerja menggunakan tampilan konsol.

Ubuntu *server* cocok digunakan untuk pembuatan *Web server*, *Mail server*, *File server* ataupun jenis *server* lainnya. Ubuntu *server* juga baik digunakan untuk pembuatan *server cloud computing*. Ubuntu sendiri memiliki servis tersendiri untuk cloud computing yang disebut dengan *Metal as a Service* (MaaS).

Berbeda dengan beberapa sistem operasi linux lainnya, ubuntu memiliki dukungan komunitas yang sangat besar jika dibandingkan dengan sistem operasi linux lainnya dan dukungan dari perusahaan canonical.

4.1.1.2 OpenLDAP + phpLDAPAdmin

OpenLDAP digunakan dalam manajemen user sedangkan phpLDAPAdmin digunakan untuk melakukan manajemen data yang berada didalam OpenLDAP. OpenLDAP bersifat *opensource* sehingga pengguna dapat merubah kode yang ada di dalam OpenLDAP agar sesuai dengan apa yang diinginkan oleh pengguna.

Penulis mempunyai beberapa pilihan untuk manajemen *user* sehingga dapat terintegrasi dengan *cloud storage* yang ada, diantara lainnya adalah *Active Directory server*. *Active Directory server* memiliki fungsionalitas yang sama dengan OpenLDAP *server*. Berikut ini adalah perbandingan antara OpenLDAP *server* dengan *Active Directory Server*, lihat tabel 4.2

Tabel 4.2 Perbandingan OpenLDAP dan *Active Directory*

Fungsi	OpenLDAP	<i>Active Directory Server</i>
Manajemen User	v	v
Open Source	v	-
Fleksibilitas	v	-
Dukungan Sistem Operasi (Ubuntu)	v	-

Dari hasil perbandingan diatas antara OpenLDAP dan *Active Directory server* maka OpenLDAP dirasa oleh penulis lebih cocok untuk manajemen *user* didalam *cloud storage* yang akan dibangun di jurusan informatika dikarenakan fleksibilitas dalam pembuatan skema untuk data user di dalam OpenLDAP lebih dapat dikustomisasi sesuai keinginan pengguna kemudian OpenLDAP didukung oleh penggunaan sistem operasi ubuntu yg digunakan dalam pembuatan *cloud storage* ini.

4.1.1.3 XFS File Sistem

Dalam komputasi, sistem file adalah jenis penyimpanan data yang dapat digunakan untuk menyimpan, mengambil dan memperbaharui sebuah set file. Istilah ini mengacu kedalam salah satu struktur data abstrak yang digunakan

dalam mendefinisikan file, perangkat lunak ataupun *firmware*. Sistem file mengelola akses isi file dan metadata tentang file yang diakses oleh user maupun oleh sistem. Hal ini berfungsi untuk mengatur ruang penyimpanan data, kehandalan, efisiensi dan tuning yang berkaitan dengan media penyimpanan fisik.

Dalam pembuatan *cloud storage* ini penulis memiliki beberapa pilihan file sistem yang dapat digunakan. Kriteria yang dicari dari file sistem ini adalah kecepatan pengaksesan file didalam *cloud storage* serta kemampuan file sistem dalam menanggapi data yang besar dan banyak.

Tabel 4.3 Perbandingan fitur XFS dengan Unix File System (UFS), Veritas File System (VxFS), dan Microsoft NT File System (NTFS).

Feature	XFS	UFS	VxFS	NTFS
Max FS Size	18 million TB	1 TB	1 TB	2 TB
Max File Size	9 million TB	1 TB	1 TB	1 TB
File Space Allocation	Extents	Blocks	Extents	Extents
Max. Extent Size	4 GB	NA	64 MB	Undoc'd
Free Space Mgmt	Free extents organized by B+ trees	Bitmap per cylinder grp	Bitmap per allocation unit	Single bitmap
Variable Block Size	512 Bytes to 64 KB	4KB or 8KB	-	512 Bytes to 64KB (4KB with compression)
Sparse File Support	Yes	Yes	No	NT 5.0
Directory Organization	B+ Tree	Linear	Hashed	B+ Tree
Inode allocation	Dynamic	Static	Dynamic	Dynamic
Crash Recovery	Asynch. Journal	Fsck*	Synch. Journal	Synch. Journal
Maximum Performance	7GB/sec, 4GB/sec (single file)	Not Available	1GB/sec	Not Available

Pada tabel 4.3 terdapat perbandingan terhadap *file sistem* yang dapat digunakan dalam membangun *cloud storage* yang akan digunakan di Jurusan Informatika. Dari tabel diatas sistem file XFS memiliki kecepatan akses data sebesar 7GB/detik atau 4GB/detik pada sebuah file, sedangkan VxFS memiliki kecepatan 1GB/detik dalam pengaksesan sebuah file. Kemudian sistem file XFS memiliki maksimal file sistem sebesar 18 juta TB, maksimum file sistem ini terbesar jika dibandingkan dengan UFS dan VxFS yang hanya memiliki maksimal 1 TB dan NTFS memiliki 2 TB. Kemudian XFS file sistem mampu menangani data sebesar 9 juta TB pada sebuah file tunggal, hal ini sangat jauh jika dibandingkan dengan UFS, VxFS dan NTFS yang hanya mampu menangani data 1 TB pada sebuah file tunggal.

Dari hasil perbandingan diatas maka penulis memutuskan untuk menggunakan sistem file XFS untuk *cloud storage* yang akan dibangun dikarenakan kecepatan untuk pengaksesan data lebih cepat dan besar file yang dapat ditangani lebih besar dari UFS, VxFS dan NTFS.

4.1.1.4 Seafile

Modul terpenting dari pembuatan *cloud storage* ini adalah *backend* dari *cloud storage* itu sendiri dan teknologi apa yang akan digunakan. Perangkat lunak untuk *backend cloud storage* memiliki kelebihan dan kekurangan pada masing-masing fitur yang disediakannya. Penulis memiliki beberapa pilihan untuk teknologi backend yang dapat digunakan dalam pembangunan *cloud storage* ini. Beberapa *backend* yang menjadi pertimbangan penulis adalah Dropbox, Owncloud, Seafile dan Swift. Masing-masing dari teknologi yang dipaparkan oleh penulis memiliki kelebihan dan kekurangan pada masing-masing kriteria yang dibutuhkan dalam pembuatan cloud storage untuk Jurusan Informatika Universitas Islam Indonesia. Beberapa kriteria yang dibutuhkan penulis sesuai dengan hasil analisa kebutuhan adalah dapat dibuat *private cloud*, mempunyai fitur sinkronisasi, memiliki *native app* dan perangkat kolaborasi. Pada tabel 4.4 terdapat hasil perbandingan dari teknologi *backend cloud storage* yang dapat digunakan.

Tabel 4.4 Perbandingan *backend cloud storage*

Fungsi	Dropbox	Owncloud	Seafile	Swift
<i>Private Cloud</i>	–	v	v	v
Sinkronisasi	v	v	v	–
Deskop App/Native App	v	v	v	–
<i>Collaboration Tool</i>	–	v (terbatas)	v	–
Toleransi kegagalan	v	–	–	v
<i>Open Source</i>	–	v	v	v

Berdasarkan perbandingan keempat teknologi yang pernah dicoba oleh penulis dapat diambil kesimpulan bahwa Seafile dan Owncloud memiliki fitur yang dibutuhkan sesuai dengan analisa kebutuhan yang sudah dirumuskan pada bab sebelumnya. Pada Owncloud fitur kolaborasi terbatas dan tidak memiliki fitur grup dan *online collaboration tools*, sedangkan seafile mempunyai dukungan terhadap fitur tersebut.

Seafile dapat memberikan solusi atas analisa kebutuhan sistem yang sudah ditulis pada bab sebelumnya. Seafile dapat dibuat menjadi *private cloud*, mempunyai fitur sinkronisasi dengan menggunakan *desktop/native app* dan dapat melakukan kolaborasi secara *online* dengan *user* lain (editing dokumen dan fitur kolaborasi lainnya).

4.2 Tahapan Implementasi Perangkat Lunak

Dalam pengaplikasian perangkat lunak ini, terdapat beberapa tahap yang terdiri dari:

- a. Instalasi dan konfigurasi Seafile server

Pada tahap ini dilakukan konfigurasi perangkat lunak Seafile yang bertujuan sebagai penyimpanan objek data dan perangkat lunak untuk melakukan kolaborasi.

b. Instalasi OpenLDAP dan phpLDAPadmin

Pada tahap ini dilakukan instalasi dan konfigurasi pada sisi *server* untuk membangun sebuah *server* untuk otentikasi data.

c. Instalasi Mozilla Firefox

Instalasi web browser dilakukan untuk keperluan konfigurasi *swauth*. Dengan dukungan *multiplatform*, kemudahan penggunaan dan kesederhanaan tampilan pemilihan Firefox sebagai web browser dilakukan.

d. Instalasi Putty

Pada tahap ini dilakukan instalasi *remote manager* putty yang berfungsi sebagai media melakukan konfigurasi *server* secara *remote*.

4.3 Implementasi Hasil Perancangan

4.3.1 Instalasi dan konfigurasi Seafile server

1. Untuk melakukan perubahan atau penambahan konfigurasi pada *server*, yang perlu diperhatikan adalah cara melakukan login ke *server*. Di tahap ini akses masuk ke *server* dilakukan melalui remote via terminal ssh yang difasilitasi oleh Putty.
2. Konfigurasi Seafile *server* dilakukan pada computer ubuntu *server* yang beralamatkan IP 192.168.69.69. Setelah berhasil login kedalam *server*, langkah awal yang dilakukan untuk manajemen 1 buah harddisk berukuran 1 TB yaitu membuat partisi pada harddisk dengan menggunakan xfs *file system*. Dipilih xfs file system dikarenakan xfs file system sangat bagus untuk melakukan handling objek yang berukuran besar dan memiliki *extended attributes (xattr)* yang diperlukan oleh objek

didalam sebuah file system serta XFS file system mampu bekerja baik pada saat data yang ditransfer berjumlah banyak. Lihat tabel 4.1

Proses pembuatan partisi ini dapat dilakukan ketika melakukan instalasi sistem operasi ataupun menggunakan perintah `mkfs.xfs` didalam sistem operasi ubuntu. Tahap selanjutnya adalah melakukan proses “*mounting*” harddisk dengan partisi xfs tadi sehingga dapat dikenali dan digunakan oleh sistem operasi ubuntu *server* (lihat Gambar 4.2).

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda1 during installation
# UUID=cae2b0d9-74fb-4187-8266-b6b158b2cdb7 /
# /opt was on /dev/sdb1 during installation
# UUID=fb67a8e7-e188-448a-9458-71bba757faae /opt
# swap was on /dev/sda2 during installation
# UUID=399c6eb3-7f63-4963-84a3-b1c869293fb5 none
```

<file system>	<mount point>	<type>	<options>	<dump>	<pass>
ext4	/	errors=remount-ro	0	1	
xfs	/opt	defaults	0	2	
swap	sw		0	0	

Gambar 4.2 Konfigurasi “*mounting*” partisi pada fstab

Dari Gambar 4.2 terlihat bahwa partisi xfs (/dev/sdb1) telah di “*mounting*” dalam direktori /opt dimana nantinya direktori ini akan menjadi tempat penyimpanan data dari *user* melewati *Seafiler server*.

- Setelah pembuatan partisi dan pembuatan mount point hal yang menjadi perhatian adalah memastikan bahwa mount point tersebut dapat digunakan oleh admin untuk meletakkan data *Seafiler* dalam direktori di dalam /opt/objectdata (lihat Gambar 4.3).

```
root@azalea:/home/lupin# ls -l /opt
total 4
drwxr-xr-x 8 lupin lupin 4096 Jul  4 10:47 objectdata
```

Gambar 4.3 Hak akses pada /opt/objectdata

Dari Gambar 4.3 dapat diketahui bahwa /opt/objectdata memiliki hak akses untuk *user* dan group lupin:lupin, hal ini harus diatur

dikarenakan Seafiler server tidak mengizinkan administrator untuk menjalankan instalasi Seafiler server dalam keadaan root.

4. Tahap selanjutnya adalah melakukan instalasi Seafiler kedalam ubuntu server yang telah disiapkan sebelumnya. Instalasi Seafiler server ini akan dipandu dengan bantuan script yang sudah disediakan oleh Seafiler server sendiri (lihat Gambar 4.4).

```

Your seafiler users would see this name in their seafiler client.
You can use a-z, A-Z, 0-9, _ and -, and the length should be 3 ~ 15
[server name]: azalea

What is the ip or domain of this server?
For example, www.mycompany.com, or, 192.168.1.101

[This server's ip or domain]: 192.168.69.69

What tcp port do you want to use for cnet server?
10001 is the recommended port.
[default: 10001 ]

Where do you want to put your seafiler data?
Note: Please use a volume with enough free space.
[default: /home/lupin/seafiler-data ] /opt/objectdata

What tcp port do you want to use for seafiler server?
12001 is the recommended port.
[default: 12001 ]

What tcp port do you want to use for seafiler httpserver?
8082 is the recommended port.
[default: 8082 ]

This is your config information:

server name:          azalea
server ip/domain:     192.168.69.69
server port:          10001
seafiler data dir:    /opt/objectdata
seafiler port:        12001
httpserver port:      8082

If you are OK with these configuration, press [ENTER] to continue.

```

Gambar 4.4 Instalasi Seafiler server

Dari Gambar 4.4 penulis melakukan instalasi dengan informasi sebagai berikut:

- Server name : azalea
- Server ip/domain : 192.168.69.69

- Server port : 10001
 - Seafile data dir : /opt/objectdata
 - Seafile port : 12001
 - http server port : 8082
5. Setelah melakukan instalasi Seafile server maka Seafile sudah dapat digunakan namun belum memiliki web interface. Tahap selanjutnya adalah instalasi seahub yang berfungsi sebagai Seafile server web interface yang berfungsi sebagai *user control panel*. Hal yang perlu diperhatikan pada tahapan ini adalah alamat email yang nantinya akan digunakan oleh administrator untuk melakukan login (lihat Gambar 4.5).

```

-----
Seahub is the web interface for seafile server.
Now let's setup seahub configuration. Press [ENTER] to continue
-----

Please specify the email address and password for seahub administrator.
You would use them to login as admin on your seahub website.

Please specify the email address for seahub administrator:
[seahub admin email]: mimin@informatics.uil.ac.id

Please specify the passwd you want to use for seahub administrator:
[seahub admin password]:
Please ensure the passwd again:
[seahub admin password again]:

This is your seahub admin username/password

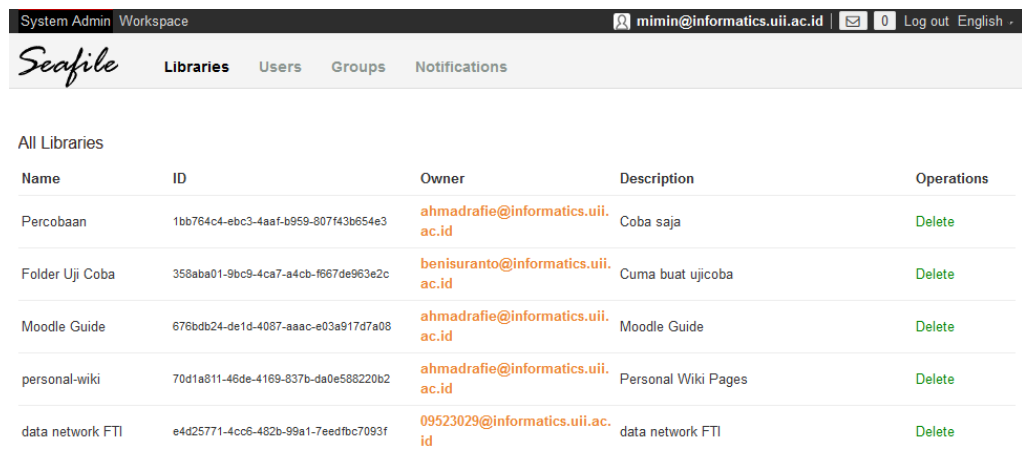
admin user name:      mimin@informatics.uil.ac.id
admin password:      *****

If you are OK with these configuration, press [ENTER] to continue.

```

Gambar 4.5 Instalasi Seahub

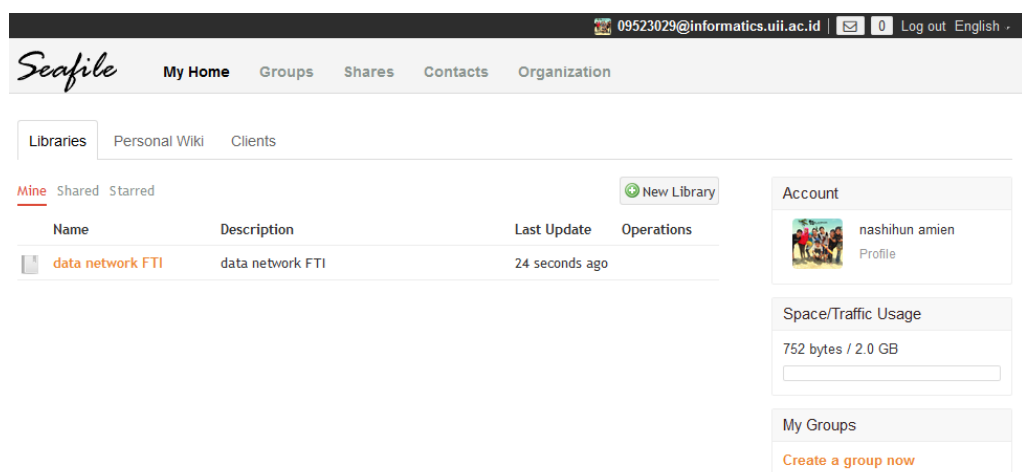
Dalam penggunaan Seafile, *user* membutuhkan akun yang berisi alamat email (diatur dalam LDAP server) yang nantinya akan digunakan untuk login kedalam *user control panel*. Untuk login kedalam server Seafile *user* akan melewati seahub sebagai *front-end* dengan melakukan akses kedalam IP server yang terletak pada alamat 192.168.69.69 dengan menggunakan port 8000.



The screenshot shows the 'System Admin' workspace of Seafile. The user 'mimin@informatics.uui.ac.id' is logged in. The 'Libraries' tab is selected, displaying a table of all libraries.

Name	ID	Owner	Description	Operations
Percobaan	1bb764c4-ebc3-4aaf-b959-807f43b654e3	ahmadrafie@informatics.uui.ac.id	Coba saja	Delete
Folder Uji Coba	358aba01-9bc9-4ca7-a4cb-f667de963e2c	benisuranto@informatics.uui.ac.id	Cuma buat ujicoba	Delete
Moodle Guide	676bdb24-de1d-4087-aaac-e03a917d7a08	ahmadrafie@informatics.uui.ac.id	Moodle Guide	Delete
personal-wiki	70d1a811-46de-4169-837b-da0e588220b2	ahmadrafie@informatics.uui.ac.id	Personal Wiki Pages	Delete
data network FTI	e4d25771-4cc6-482b-99a1-7eedfbc7093f	09523029@informatics.uui.ac.id	data network FTI	Delete

Gambar 4.6 Admin Control Panel



The screenshot shows the 'My Home' workspace of Seafile. The user '09523029@informatics.uui.ac.id' is logged in. The 'Libraries' tab is selected, displaying a table of the user's libraries. The right sidebar shows account information, space/traffic usage, and a link to create a group.

Name	Description	Last Update	Operations
data network FTI	data network FTI	24 seconds ago	

Account
 Profile: nashihun amien
 Space/Traffic Usage: 752 bytes / 2.0 GB
 My Groups: Create a group now

Gambar 4.7 User control panel

Pada Gambar 4.6 dan Gambar 4.7 kita dapat melihat bahwa *Control Panel* dari admin dan *user* biasa memiliki fitur yang sama, yang membedakan adalah pada admin *Control Panel* terdapat fitur untuk melihat seluruh aktivitas *user* yang ada dan dapat melakukan pengaturan terhadap kuota data storage yang dimiliki *user*.

4.3.2 Instalasi dan Konfigurasi OpenLDAP

1. OpenLDAP pada sistem ini terdapat didalam *server* yang berbeda dari *server* Seafile hal ini dilakukan agar *server* dapat mengerjakan pekerjaan

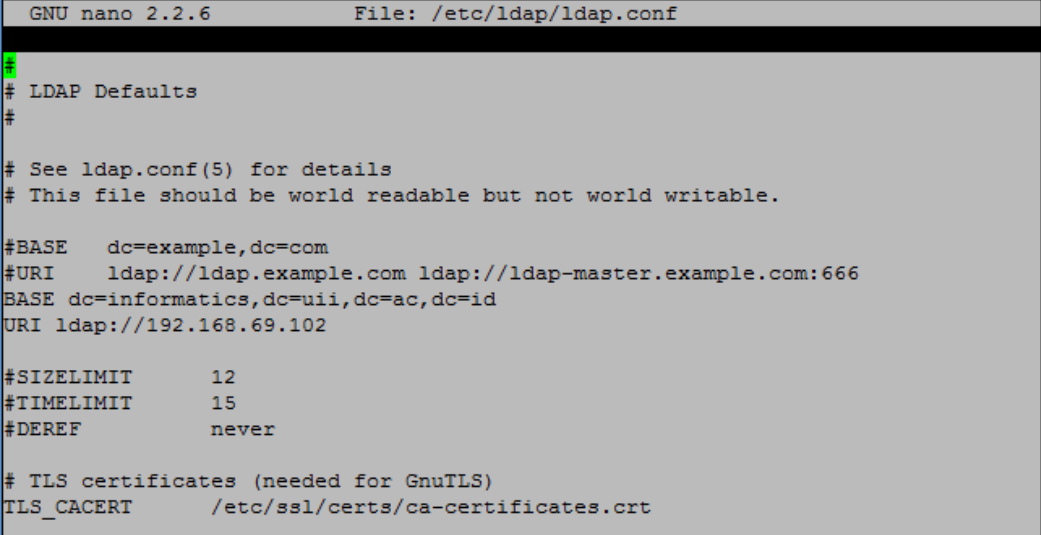
yang dijalankan secara spesifik. OpenLDAP menggunakan sistem operasi ubuntu server dengan alamat IP 192.168.69.102.

2. Instalasi OpenLDAP server menggunakan perintah apt-get

```
root@pandora-box:/home/lupin# apt-get install slapd
ldap-utils
```

Gambar 4.8 Instalasi OpenLDAP

Hal yang perlu diperhatikan dalam pembuatan LDAP server ini adalah struktur hierarki domain. Untuk menjalankan OpenLDAP kita memerlukan inisialisasi domain untuk jurusan informatika.



```
GNU nano 2.2.6      File: /etc/ldap/ldap.conf
#
# LDAP Defaults
#
# See ldap.conf(5) for details
# This file should be world readable but not world writable.
#BASE    dc=example,dc=com
#URI      ldap://ldap.example.com ldap://ldap-master.example.com:666
BASE dc=informatics,dc=uui,dc=ac,dc=id
URI ldap://192.168.69.102

#SIZELIMIT      12
#TIMELIMIT      15
#DEREF          never

# TLS certificates (needed for GnuTLS)
TLS_CACERT      /etc/ssl/certs/ca-certificates.crt
```

Gambar 4.9 Konfigurasi domain dan IP pada LDAP server

Pada gambar 4.9 alamat ip dikonfigurasi pada alamat dimana server LDAP ini dijalankan. Yang menjadi perhatian adalah konfigurasi nama domain yang akan menentukan struktur hierarki domain pada LDAP server yang dibuat. Nama domain yang akan dibuat adalah `informatics.uui.ac.id`, maka penulisan dalam LDAP server mengikuti format penulisan yang diketahui oleh LDAP server itu sendiri yaitu `dc=informatics,dc=uui,dc=ac,dc=id`. Untuk pengecekan LDAP server dapat menggunakan perintah `ldapsearch -x`

3. Tahap selanjutnya adalah melakukan instalasi phpldapadmin yang akan digunakan untuk melakukan manajemen LDAP server. Instalasi phpldapadmin dapat menggunakan perintah sebagai berikut:

```
root@pandora-box:/home/lupin# apt-get install phpldapadmin
```

Gambar 4.10 Instalasi phpLDAPAdmin

Setelah tahap instalasi selesai dilakukan, hal yang perlu dilakukan adalah melakukan konfigurasi agar phpldapadmin dapat berjalan dengan domain yang telah dibuat diatas. Letak konfigurasi file berada didalam direktori /etc/phpldapdamin/config.php. Beberapa parameter yang perlu didefinisikan dalam konfigurasi phpldapadmin adalah sebagai berikut

```
$servers = new Datastore();
$servers->newServer('ldap_pla');
$servers->setValue('server','name','Informatics LDAP
Server');
$servers->setValue('server','host','192.168.69.102');
$servers-
>setValue('server','base',array(dc=informatics,dc=uii,dc=ac,
dc=id));
$servers-
>setValue('login','bind_id','cn=admin,dc=informatics,dc=uii,
dc=ac,dc=id);
```

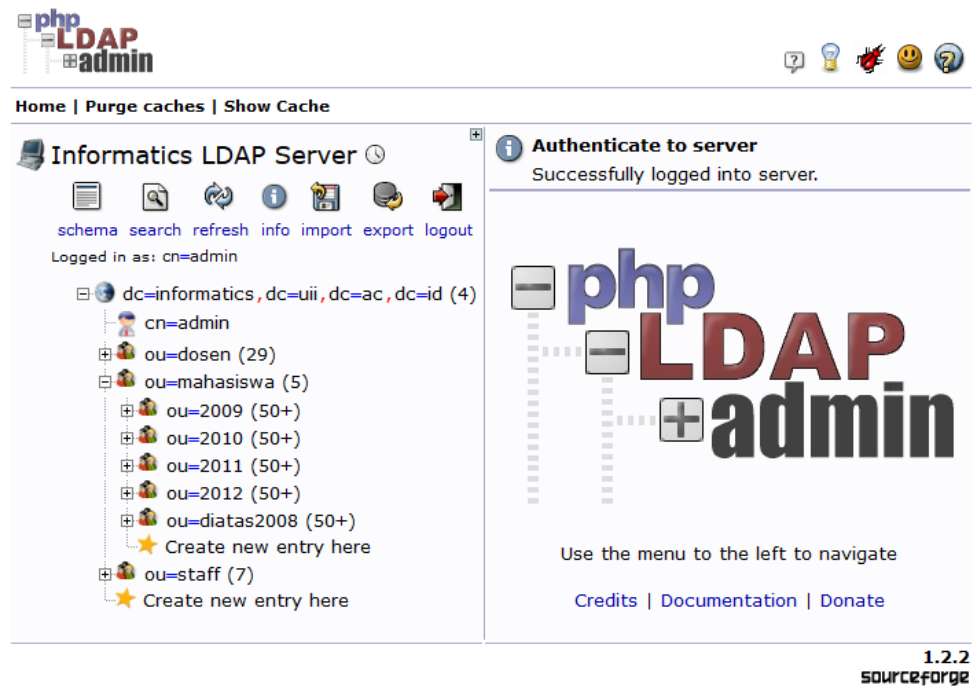
Gambar 4.11 Isi konfigurasi /etc/phpldapdamin/config.php

Phpldapadmin menggunakan port 389 dalam menjalankan service-nya, yang perlu diperhatikan adalah firewall mengijinkan port tersebut diakses atau tidak. Untuk memastikan bahwa port dapat digunakan, langkah selanjutnya adalah memberikan hak akses kepada port 389 agar dapat digunakan dengan menggunakan perintah berikut :

```
root@pandora-box:/home/lupin# ufw allow 389
```

Gambar 4.12 Konfigurasi *firewall*

4. Setelah phpldapadmin berhasil dibuat (lihat Gambar 4.9) kemudian tahap selanjutnya adalah membuat akun untuk semua *user* yang berada dilingkungan jurusan teknik informatika.



Gambar 4.13 phpldapadmin Control Panel

Pembuatan akun terbagi menjadi beberap bagian organisasi unit yaitu dosen staff dan mahasiswa. Kemudian pada bagian mahasiswa dipisahkan berdasarkan tahun angkatan agar mempermudah dalam manajemen *user*.

Pada tahap ini pembuatan akun *user* berdasarkan file csv dari Unisys yang kemudian dikonversikan kedalam format ldif, yaitu format yang dikenali oleh LDAP *server*. Contoh penulisan format csv yang akan dikonversikan kedalam format ldif adalah sebagai berikut:

```
dn,objectclass,objectclass,cn,givenname,mail,sn,uid,userpass
word
"cn=inf
09523007,ou=2009,ou=mahasiswa,dc=informatics,dc=uii,dc=ac,dc
```

```
=id",inetOrgPerson,top,9523007,inf,09523007@informatics.iii.
ac.id,maikynofriyanto,9523007, *****
"cn=inf
09523008,ou=2009,ou=mahasiswa,dc=informatics,dc=uii,dc=ac,dc
=id",inetOrgPerson,top,9523008,inf,09523008@informatics.iii.
ac.id,setiyosonugroho,9523008,*****
```

Gambar 4.14 Konversi CSV ke LDIF

Kemudian dengan format csv ini maka akan kita konversikan kedalam format ldif yang dikenali oleh LDAP server, sehingga nantinya file tersebut akan mudah untuk melakukan proses import data kedalam LDAP server.

5. Pada tahap ini setelah file csv yang berisi data seluruh user yang berada didalam jurusan informatika telah selesai dibuat, maka tahap selanjutnya adalah melakukan konversi data dari format file csv kedalam format ldif yang dikenali oleh LDAP server. Konversi data dilakukan dengan menggunakan bantuan perangkat lunak CSV2LDIF. Cara melakukan konversi csv kedalam ldif dapat dilakukan dengan menggunakan perintah sebagai berikut:

```
Csv2ldif.exe /i=nama_file.csv /o=nama_file.ldif
```

Gambar 4.15 Konversi csv menggunakan csv2ldif.exe

Setelah konversi data selesai dilakukan maka akan terbentuklah sebuah data baru hasil konversi data dari file csv tersebut. Format data ldif yang terbentuk adalah sebagai berikut:

```
dn:
cn=9523007,ou=2009,ou=mahasiswa,dc=informatics,dc=uii,dc=ac,
dc=id
cn: 9523007
givenname: inf
mail: 09523007@informatics.iii.ac.id
objectclass: top
objectclass: inetOrgPerson
sn: maikynofriyanto
uid: 9523007
userpassword: *****
```

```

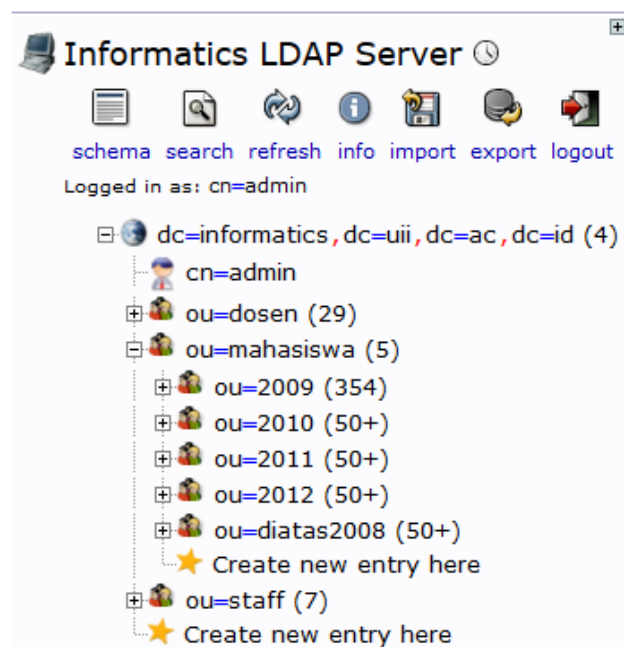
dn:
cn=9523008,ou=2009,ou=mahasiswa,dc=informatics,dc=uii,dc=ac,
dc=id
cn: 9523008
givenname: inf
mail: 09523008@informatics.uui.ac.id
objectclass: top
objectclass: inetOrgPerson
sn: setiyosonugroho
uid: 9523008
userpassword: *****

```

Gambar 4.16 LDIF Format untuk LDAP server

Setelah data dengan format ldif berhasil dibuat maka tahap selanjutnya adalah melakukan export data kedalam LDAP *server* melalui phpldapadmin yang memiliki fitur export didalamnya.

Setelah melakukan export data kedalam LDAP *server* maka akan muncul didalam tree viewer phpldapadmin *user* dan unit organisasi yang berada diatasnya sesuai dengan nama domain yang telah dibuat diatas (lihat Gambar 4.10).



Gambar 4.17 Hasil export data yang dikonversi

Hal yang perlu diperhatikan selanjutnya adalah mengkoneksikan LDAP server dengan Seafile agar Seafile dapat terintegrasi dengan LDAP server dalam manajemen akun *user*. Hal tersebut dapat diatur didalam file konfigurasi Seafile yang terletak didalam file `ccnet.conf`.

```
[LDAP]
HOST = ldap://192.168.69.102/
BASE = dc=informatics,dc=uii,dc=ac,dc=id
USER_DN = cn=admin,dc=informatics,dc=uii,dc=ac,dc=id
PASSWORD = *****
LOGIN_ATTR = mail
```

Gambar 4.18 Konfigurasi tambahan dalam `ccnet.conf`

Dapat dilihat pada Gambar 4.11 terdapat parameter `LOGIN_ATTR = mail`, hal tersebut menunjukkan bahwa *user* hanya dapat login menggunakan alamat email yang terdaftar didalam LDAP server.

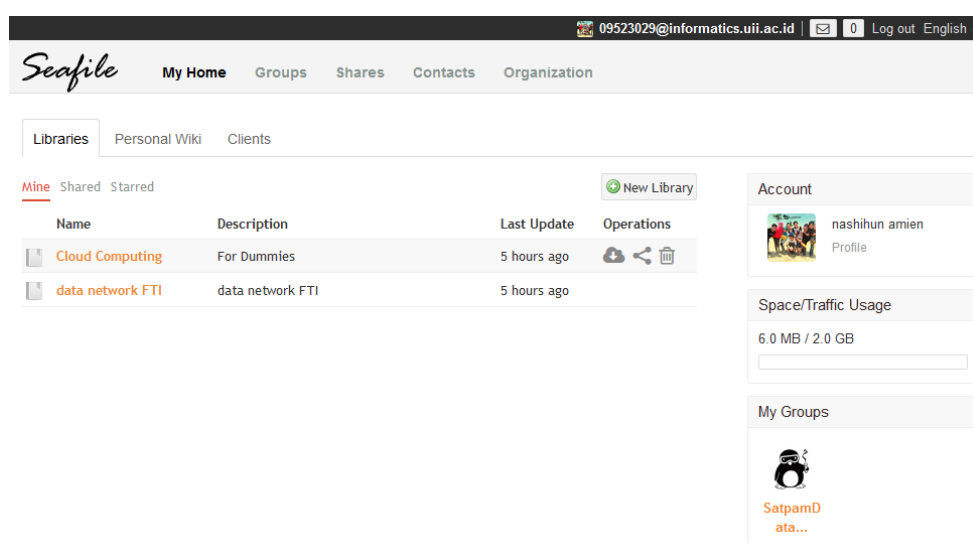
4.4 Pengujian Hasil Implementasi

4.4.1 Pengujian fungsionalitas layanan Seafile server

Menjadi *user-oriented model* memiliki arti bahwa informasi yang diberikan pada model ini dapat memberikan manfaat untuk sebagian pengguna jaringan Jurusan Informatika UII. Ada beberapa skenario yang dapat membantu menjelaskan bagaimana pengguna dapat menggunakan sistem ini dan mendapatkan manfaat dari hal tersebut. Setidaknya ada dua sampel yang dapat dijelaskan lebih lanjut dibawah.

Yang pertama adalah penggunaan layanan penyimpanan data. Ketika sebuah data diupload kedalam sistem *cloud storage* ini maka data secara otomatis akan disimpan kedalam sebuah direktori data yang terstruktur dan ditambah dengan enkripsi data yang aman. Dengan adanya hal ini memungkinkan untuk tidak terjadinya data digunakan oleh pihak-pihak yang tidak diinginkan. Kemudian adanya sinkronisasi data memungkinkan *user* menggunakan data secara offline yang kemudian dapat diperbaharui data didalam *cloud storage* dengan mensinkronisasikannya sewaktu-waktu.

Penggunaan lain dari sistem *cloud storage* ini adalah adanya fitur kolaborasi yang memungkinkan *user* untuk saling berbagi sumberdaya dan mengerjakan pekerjaan secara langsung didalam *cloud storage*. *User* dapat membuat dokumen langsung didalam *cloud storage* dengan menggunakan word editor yang ada didalam *cloud storage*. Kemudian fitur group membuat jalannya kolaborasi akan menjadi lebih mudah dan terstruktur. Gambar 4.19 adalah tampilan *control panel* dari *user* dimana terdapat beberapa menu seperti kontak, grup, sharing dan organisasi. Gambar 4.20 menunjukkan fitur sharing, dimana fitur tersebut dapat digunakan oleh *user* untuk berbagi sumber daya serta melakukan kolaborasi. Pada Gambar 4.21 dapat dilihat bahwa *user* dapat memiliki kontak dengan *user* lain dimana nantinya kontak ini berfungsi untuk menunjukkan alamat email yang akan digunakan untuk berbagi file ataupun mengundang *user* kedalam sebuah grup. Gambar 4.22 dan 4.24 menunjukkan fitur group yang dapat digunakan untuk melakukan diskusi antar sesama *user* didalamnya. Gambar 4.23 adalah kontrol panel dari admin grup berfungsi untuk menambah *user* kedalam grup atau sebaliknya. Gambar 4.25 berikut ini adalah tampilan dari *Seafile client* yang digunakan oleh *user* untuk mengunggah/mengunduh data serta untuk melakukan kolaborasi dengan *user* lain dan melakukan sinkronisasi dengan *Seafile server*.



Gambar 4.19 User Control Panel

The screenshot shows the Seafile interface with the 'Shares' tab selected. The top navigation bar includes 'My Home', 'Groups', 'Shares', 'Contacts', and 'Organization'. The main content area is titled 'Shared Libraries' and contains a table with the following data:

Name	Share To	Permission	Description	Operations
Cloud Computing	all members	Read-Only	For Dummies	Unshare
data network FTI	SatpamDatacenter	Read-Write	data network FTI	Unshare

On the right, a 'Tips' box states: 'If you don't want others to change the contents of your shared libraries, you can set the share permission to "read-only" when sharing them.'

Gambar 4.20 Fitur sharing file dan dokumen untuk berkolaborasi

The screenshot shows the Seafile interface with the 'Contacts' tab selected. The top navigation bar includes 'My Home', 'Groups', 'Shares', 'Contacts', and 'Organization'. The main content area is titled 'Contacts' and features an 'Add Contacts' button. Below it is a table with the following data:

Email	Name	Note	Operations
ahmadrafie@informatics.uui.ac.id			Edit Delete
09523008@informatics.uui.ac.id			Edit Delete
09523071@informatics.uui.ac.id			Edit Delete

On the right, a 'Tips' box states: 'When you share libraries or file links to a friend, his/her email will be added to contacts.'

Gambar 4.21 Fitur contacts didalam Seafile

The screenshot shows the Seafile interface with the 'Groups' tab selected. The top navigation bar includes 'My Home', 'Groups', 'Shares', 'Contacts', and 'Organization'. The main content area is titled 'My Groups' and features a 'New Group' button. Below it, a group named 'SatpamDatacenter' is displayed with the following details:

SatpamDatacenter
 Creator: 09523029@informatics.uui.ac.id
 Time: 2013-07-03 03:19:48

On the right, a 'Tips' box states: 'After creating a group, you can add members and share libraries into it. You can also share libraries into groups that you're not in.'

Gambar 4.22 Fitur group sebagai tempat diskusi user

The screenshot shows the Seafile web interface. At the top, there's a header with the Seafile logo and navigation links: My Home, Groups, Shares, Contacts, and Organization. Below this, there's a sub-header with tabs: Libraries, Discussion, Wiki, Members, and Admin. The 'Libraries' tab is selected. On the right, there's a user profile for 'SatpamDatacenter' with a penguin icon and the name 'nashihun amien'. Below the profile, there's a 'Basic Info' section showing the user's creation date as '2013-07-03'. To the left of the profile, there's a table of libraries. The table has columns: Name, Description, Last Update, Shared By, and Operations. One library is listed: 'data network FTI' with a description 'data network FTI', last update '10 hours ago', and shared by 'nashihun amien'. There's also a 'New Library' button.

Name	Description	Last Update	Shared By	Operations
data network FTI	data network FTI	10 hours ago	nashihun amien	

Gambar 4.23 Control Panel administrator group

The screenshot shows the Seafile web interface with the 'Discussion' tab selected. On the left, there's a discussion area with a text input field 'Add a discussion...' and a button to post. Below it, there's a post from 'nashihun amien' with the text 'Hello world!' and 'Vi veri veniversum vivus vici'. There's also a text input field 'Add a reply...' and a button to reply. On the right, there's a 'Members' section listing the group members: 'ahmadrafie', '09523071', 'nashihun amien', and '09523008'.

Gambar 4.24 Fitur diskusi didalam group

The screenshot shows the Seafile client application interface. At the top, there's a dark header with icons for Libraries, Settings, and Help, and a language selector '中文'. Below the header, there's a section titled 'All Libraries'. It contains a table with columns: Name, Action, Description, Server, Modify Time, and State. One library is listed: 'data network FTI' with an 'Open' action, description 'data network FTI', server 'azalea', modify time '2013-07-03 01:16:11', and state 'Synchronized'.

Name	Action	Description	Server	Modify Time	State
data network FTI	Open	data network FTI	azalea	2013-07-03 01:16:11	Synchronized

Gambar 4.25 Aplikasi Seafile client

testing.seaf's Version History

Tip: a new version will be generated after each modification, and you can restore the file to a previous version.

Current Path: [Cloud Computing / testing.seaf](#)

Time	Last Modifier	Size	Operations
1 minute ago	 nashihun amien	148 bytes	Restore Download View
1 minute ago	 09523008	158 bytes	Restore Download View
2 minutes ago	 09523008	101 bytes	Restore Download View
4 minutes ago	 nashihun amien	59 bytes	Restore Download View
5 minutes ago	 nashihun amien	0 bytes	Restore Download View

Gambar 4.26 Fitur online collaboration tool

Berbeda dengan sistem *cloud storage* ini, perangkat lunak seperti Dropbox dan Owncloud, seperti yang disebutkan sebelumnya fungsionalitas dari kedua perangkat lunak tersebut hanya terbatas pada penyimpanan data. Sebagai contoh, Dropbox tidak dapat dibuat menjadi sebuah *private cloud storage* maupun hybrid *cloud storage*. Kemudian pada Owncloud dapat dibuat menjadi hybrid *cloud storage* namun tidak memiliki fitur kolaborasi seperti Seafile (lihat Gambar 4.26).

Secara umum, dapat disimpulkan bahwa kegunaan dari sistem ini adalah pada peningkatan kualitas pengalaman ketika bekerja menggunakan sistem *cloud storage*. Kualitas ini dapat ditingkatkan dengan memberikan informasi yang tepat, yang benar-benar diperlukan oleh pengguna dari *cloud storage* ini. Hal ini dapat ditingkatkan dengan memiliki *cloud storage* yang terintegrasi dengan beberapa alat manajemen jaringan yang ada seperti *cloud backend* (AWS S3) dan sistem monitoring layanan.

4.4.2 Pengujian kinerja layanan Seafile server

Segala aplikasi yang berjalan di dalam sistem linux ataupun windows memiliki *service* yang disebut daemon. Service ini berjalan di belakang layar dan bisa dilihat dengan mengetikkan perintah “ps aux” (linux) atau dengan “task manager” (windows). Service-service yang dipantau yaitu Seafile server, ccnet, dan httpserver yang dijalankan oleh komputer *cloud storage*. Gambar berikut ini adalah tampilan status running program aplikasi.

```

root 1458 0.0 0.0 52092 1432 ? Ss Jul02 0:00 /home/lupin/seafile/seafile/bin/seafile-
root 1460 0.0 0.4 316060 8372 ? Ss Jul02 0:02 ccnet-server -c /home/lupin/ccnet -d -P
root 1463 0.0 0.0 0 0 ? Z Jul02 0:00 [httpserver] <defunct>
root 1464 0.0 0.1 385492 3812 ? Ss Jul02 0:00 seaf-mon -c /home/lupin/ccnet -d /opt/ob
root 1465 0.0 0.2 312240 4488 ? Ss Jul02 0:04 seaf-server -c /home/lupin/ccnet -d /opt
root 1466 0.0 0.2 847580 4444 ? Ss1 Jul02 0:00 httpserver -c /home/lupin/ccnet -d /opt/
root 1495 0.0 1.0 71120 22052 ? S Jul02 0:01 python2.7 /home/lupin/seafile/seahub/man
root 1498 0.0 1.6 97368 34068 ? S Jul02 0:05 python2.7 /home/lupin/seafile/seahub/man
root 1499 0.0 1.5 95016 31476 ? S Jul02 0:03 python2.7 /home/lupin/seafile/seahub/man
root 1500 0.0 1.7 98452 35152 ? S Jul02 0:04 python2.7 /home/lupin/seafile/seahub/man

```

Gambar 4.27 Service yang berjalan pada *cloud storage server*

Dapat terlihat melalui gambar diatas terdapat aplikasi Seafile server, ccnet serta seahub yang menandakan bahwa *service* telah berjalan dan siap untuk diuji.

4.4.2.1 Pengujian performa pengiriman data

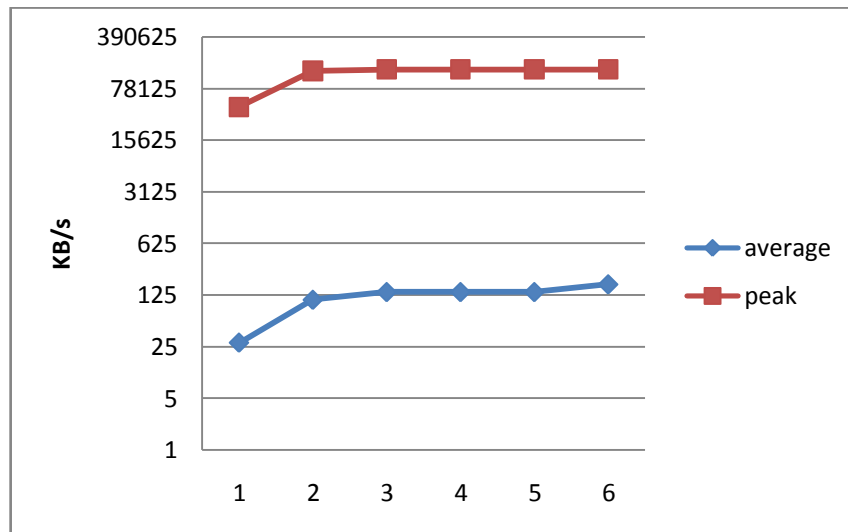
Tahap uji coba ini dilakukan dengan mengkoneksikan *server cloud storage* dengan klien computer penguji. Klien terhubung dengan interface NIC dari komputer *server* melalui perantara switch

a. Pengujian throughput unggah data

Data yang berhasil diambil dapat dilihat pada tabel 4.2 berikut ini.

Tabel 4.5 Rata-rata throughput dan peak unggah data

Unggah (KB/s)		
No	Rata-rata Throughput	peak
1	28.21	43975
2	107.9	134012.5
3	137.5	140550
4	137.5	140550
5	137.5	140550
6	175	140550



Gambar 4.27 Uji performa unggah data pada *cloud storage*

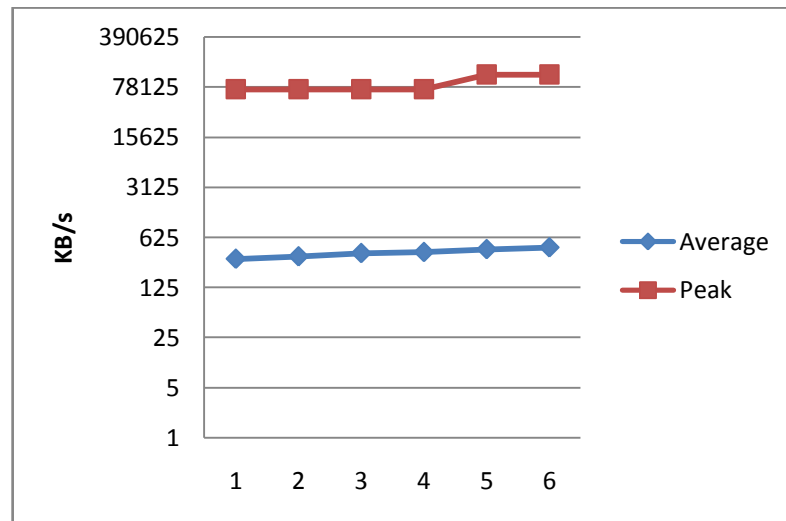
Gambar diatas menunjukkan performa rata-rata throughput dalam melakukan unggah data kedalam *server cloud storage*. Dari percobaan mengirimkan sebuah paket sebesar 500MB dari 25 komputer secara bersamaan membutuhkan waktu (*transfer time*) selama 15 menit 21,13 detik.

b. Pengujian throughput unduh data

Data yang berhasil diambil dapat dilihat pada table 4.3 berikut ini

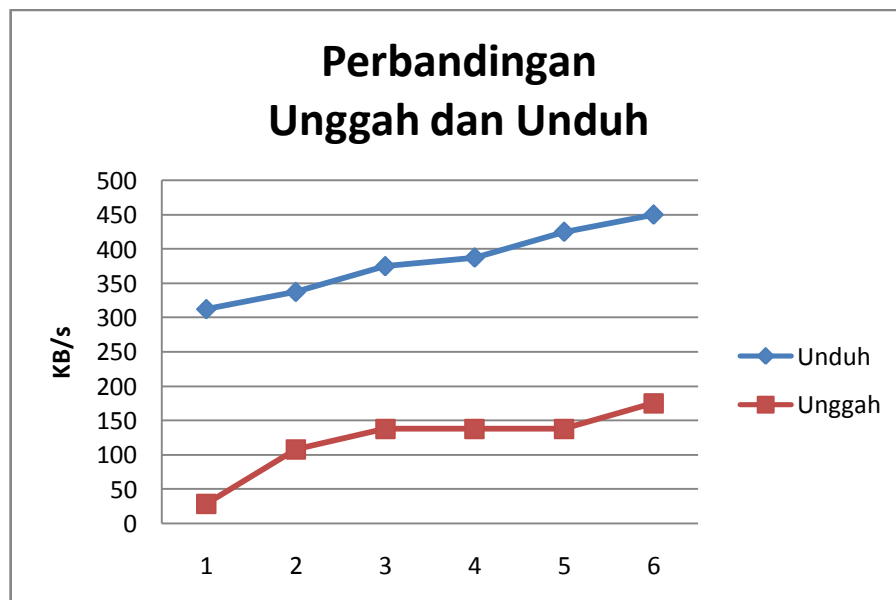
Tabel 4.6 Rata-rata throughput dan peak unduh data

Unduh (KB/s)		
No	Rata-rata Throughput	peak
1	312.5	72462.5
2	337.5	72462.5
3	375	72462.5
4	387.5	72462.5
5	425	115612.5
6	450	115612.5



Gambar 4.28 Uji performa unduh data pada *cloud storage*

Gambar diatas menunjukkan performa rata-rata throughput dalam melakukan pengunduhan data dari *server cloud storage* kedala komputer klien. Menurut hasil pengamatan setelah melakukan percobaan mengunduh data sebesar 500MB dari 25 komputer secara bersamaan membutuhkan waktu (*transfer time*) selama 18 menit 40,36 detik



Gambar 4.29 Perbandingan *throughput* saat mengunggah dan mengunduh data

Pada Gambar 4.29 menunjukkan perbandingan rata-rata *throughput* ketika melakukan percobaan mengunduh dan mengunggah data dengan data sebesar 500MB dari 25 komputer. Perbedaan rata-rata throughput yang didapat ketika user melakukan pengunduhan data sangat signifikan.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan analisis dari parameter fungsionalitas serta *throughput* dan *transfer time* dari percobaan mengunggah dan mengunduh data dari server *cloud storage* maka dapat ditarik kesimpulan sebagai berikut :

1. *Private cloud storage* dibangun dengan fitur sinkronisasi dan *sharing file* maka user akan mudah dalam melakukan aktifitas berbagi sumber daya. Dengan adanya fitur ini data akan terkirim kedalam server *private cloud storage*, sehingga jika perangkat user mengalami kegagalan dalam pengaksesan data secara lokal maka *user* dapat menggunakan file yang sudah berada didalam *private cloud storage* tersebut.
2. Fungsionalitas dari server *cloud storage* ini cukup lengkap bila dibandingkan dengan Dropbox, Owncloud, dan Swift dengan tambahan fitur kolaborasi menjadikan *private cloud storage* ini mempunyai fungsionalitas lebih jika digunakan dalam kegiatan transfer ilmu di dalam lingkungan Jurusan Informatika UII
3. Performa dari server *cloud storage* ini terbukti dengan rata-rata *transfer time* pada saat mengunggah data pada server *cloud storage* adalah 15 menit 21,13 detik. Dan pada saat mengunduh data *transfer time* menunjukkan waktu 18 menit 40,36 detik dengan besar data sebesar 500MB dari 25 user secara bersamaan. *Troughput* rata-rata pada saat mengunggah data adalah 175 KB/s dan rata-rata pada saat mengunduh data adalah 450 KB/s.

5.2 Saran

Berdasarkan hasil pengujian dan analisis data yang telah dilakukan, saran yang dapat diberikan oleh penulis untuk pembaca apabila ingin melakukan penelitian lebih lanjut tentang *private cloud storage* adalah adalah dalam

pemilihan perangkat yang digunakan baik perangkat lunak maupun keras. Selain itu perlu ditambahkan beberapa parameter pengukuran dan fitur baru sehingga penelitian terhadap kinerja dapat diketahui lebih lanjut lagi mengenai kelebihan didalamnya serta dapat memberikan layanan baru dikemudian hari.

Selain menganalisis dan memberikan fitur baru juga dapat dilakukan penelitian lebih lanjut tentang fitur toleransi kegagalan dimana nantinya ketika server sudah digunakan secara terus-menerus sistem dapat melakukan penyalinan data secara otomatis jika terjadi kegagalan sistem.

DAFTAR PUSTAKA

Voas, J., & Zhang, J. 2009. Cloud computing: New wine or just a new bottle? IEEE ITPro, 15–17.

Sun Microsystems. 2009. Introduction to cloud computing architecture. White Paper, Sun Microsystems.

Jens, F. 2008. Defining cloud services and cloud computing. [online] (<http://blogs.idc.com/ie/?p=190>. Diakses 20 Januari 2013).

Amazon Simple Storage Service (Amazon S3). 2010. [online] (<http://aws.amazon.com/s3>. Diakses pada 14 Februari 2013).

Dropbox Homepage. 2010. [online] (<http://www.dropbox.com/>. Diakses pada 22 Juni 2013).

Owncloud Homepage. 2010. [online] (<http://owncloud.org/>. Diakses pada 22 Juni 2013).

Andrew. 2013. Seafile: Robust File Synchronization And Collaboration Tools That You Can Run On Your Linux Server. [online] (<http://www.webupd8.org/2013/02/Seafile-robust-file-synchronization-and.html>. Diakses pada 15 Maret 2013).

Clayton, Chris. Standart Cloud Taxonomies and Windows Azure. 2011 [online] (<http://blogs.msdn.com/b/cclayton/archive/2011/06/07/standard-cloud-taxonomies-and-windows-azure.aspx>. Diakses 23 Juni 2013).

Trautman, Philip., & Mostek, Jim. Scalability and Performance in Modern File Systems [online]. (http://linux-xfs.sgi.com/projects/xfs/papers/xfs_white/xfs_white_paper.html. Diakses pada 23 Juni 2013).

Anonim. What is File Level Storage vs. Block Level Storage? [online] (<http://www.stonefly.com/resources/File-Level-Storage-vs-Block-Level-Storage.asp>. Diakses pada 23 Juni 2013).

Oxnevad, Susan. 2012. 6 Powerful Goodle Docs Deatures to Support the Collaborative Writing Process. [online]. (http://gettingsmart.com/2012/09/google_docs_for_collaborative_writing/, Diakses pada 23 Juni 2013).

Carter, Gerald, 2003. LDAP System Administration. [online] (<http://shop.oreilly.com/product/9781565924918.do>. Diakses tanggal 23 Juni 2013).