

## BAB II

### TINJAUAN PUSTAKA DAN DASAR TEORI

#### 2.1. Tinjauan Pustaka

Yuliyanto (2016) membuat sebuah aplikasi *Cloud Storage* sebagai *storage file*. Pada aplikasi tersebut dapat melihat, mengunggah, mengunduh, menghapus serta membagikan suatu *file*. Aplikasi tersebut merupakan aplikasi multi user sehingga dapat diakses oleh beberapa pengguna dalam waktu bersamaan. Metode yang digunakan adalah metode *Codeigniter Framework*. Di dalam aplikasi tersebut menggunakan database MySQL dan bahasa pemrograman *web*.

Wahyudi (2014) membuat sebuah aplikasi dengan memanfaatkan teknologi *cloud computing* sebagai media penyimpanan data yang dapat melakukan sinkronisasi data dari sisi *client* ke *server* maupun *server* ke *client*. Dengan adanya perancangan aplikasi tersebut dapat disimpulkan bahwa pembuatan aplikasi *cloud storage* tidak hanya dapat dibuat dengan bahasa pemrograman web tetapi bisa juga dibuat dengan menggunakan bahasa pemrograman java. Metode yang digunakan adalah metode komparasi. Di dalam aplikasi tersebut menggunakan database MySQL dan bahasa pemrograman java.

Ibrahim dan Kusnawi (2013) menganalisis dan membuat sebuah media penyimpanan data di Yayasan Salman AL-FARISI Yogyakarta dengan menggunakan perangkat lunak owncloud sebagai *cloud storage*. Dengan diterapkan sebuah layanan *cloud storage* ini dapat memudahkan semua lembaga pendidikan untuk menyimpan atau membagi *file* dengan lembaga lain. Selain itu

memberikan efektivitas dalam bekerja. Metode yang digunakan adalah metode *cloud storage*. Di dalam aplikasi tersebut menggunakan database MySQL dan bahasa pemrograman *web*.

Penelitian lain merancang sebuah layanan *private cloud storage* menggunakan ownCloud pada Program Studi Magister Ilmu Lingkungan di Universitas Sebelas Maret. Dengan diterapkan sebuah layanan *cloud storage* di institusi dapat mempermudah melakukan pertukaran informasi (*file*) jarak jauh oleh setiap staf kantor, mahasiswa dan staf pengajar selain itu dapat digunakan untuk melakukan *backup file* sehingga resiko kehilangan data dapat dihindari. Metode yang digunakan adalah metode *private cloud storage*. Di dalam aplikasi tersebut menggunakan database MySQL dan bahasa pemrograman *web* (Nugroho, 2014).

Penelitian lain juga membuat modul layanan sinkronisasi data antara perangkat lunak ownCloud dengan Google Drive sebagai *backup* data, kemudian melakukan perbandingan waktu sinkronisasi data pada perangkat lunak ownCloud dengan Google Drive sebagai *backup* data. Metode yang digunakan adalah metode sinkronisasi data satu arah. Di dalam aplikasi tersebut menggunakan database MySQL dan bahasa pemrograman *web* (Basari, 2014).

Telah pustaka terhadap penelitian-penelitian sebelumnya menunjukkan bahwa belum terdapat penelitian mengenai pengembangan aplikasi sinkronisasi data pada *server* ownCloud dengan Dropbox untuk keperluan *backup* dengan memanfaatkan Dropbox API. Metode yang digunakan adalah metode sinkronisasi satu arah. Didalam aplikasi tersebut menggunakan database MySQL dan bahasa

pemrograman *web*. Rangkuman mengenai penelitian yang telah dilakukan dan akan dilakukan seperti pada Tabel 2.1.

Tabel 2.1. Data Penelitian Mengenai Pengembangan dan Pembuatan Aplikasi *Cloud Storage*.

Penulis	Objek	Metode	Bahasa Pemrograman	Interface
Sigit Yuliyanto (2016)	Individu	<i>Codeigniter Framework</i>	<i>Web</i> dan MySQL	GUI
Robertus Nugraha Setya Wahyudi (2014)	<i>client server</i>	komparasi	Java dan MySQL	GUI
Muhammad Ibrahim dan Kusnawi (2013)	Yayasan Salman AL-FARISI Yogyakarta	<i>public cloud storage</i>	<i>Web</i> dan MySQL	GUI
Tri Adi Nugroho (2014)	Universitas Sebelas Maret	<i>private cloud storage</i>	<i>Web</i> dan MySQL	GUI
Basari (2014)	Individu	Sinkronisasi Data Satu Arah	<i>Web</i> dan MySQL	GUI
Faizal Azis (2018)	Individu	Sinkronisasi Data Satu Arah	<i>Web</i> dan MySQL	GUI

## 2.2. Dasar Teori

### 2.2.1. *Cloud Computing*

*Cloud Computing* merupakan sebuah bentuk layanan yang membuka peluang untuk dapat hadir dimanapun, memberikan kenyamanan, akses jaringan sesuai permintaan (*on-demand*) ke lokasi sumber daya komputasi terkonfigurasi (misalnya, jaringan, *server*, penyimpanan, aplikasi, dan layanan). Menurut NIST (*National Institute of Standards and Technology*) ada tiga jenis layanan dalam *cloud computing* diantaranya sebagai berikut:

#### 1. *Software as a Service*

Kemampuan yang diberikan kepada *consumer* untuk menggunakan aplikasi milik pemberi layanan yang bekerja pada infrastruktur *cloud*. Aplikasi-aplikasi tersebut dapat diakses dari perangkat *client* apapun, baik melalui *thin client interface* seperti *web-browser* (*web-based email*) atau tampilan antarmuka program.

#### 2. *Platform as a Service*

Kemampuan diberikan kepada *consumer* untuk meluncurkan kedalam infrastruktur *cloud* aplikasi-aplikasi yang *consumer* buat, menggunakan bahasa pemrograman, *libraries*, layanan, dan alat bantu yang disediakan oleh penyedia jasa layanan.

#### 3. *Infrastructure as a Service*

Kemampuan diberikan pada *consumer* untuk menentukan pengelolaan, penyimpanan, jaringan dan sumber daya lainnya yang mendasar komputasi di

mana *consumer* dapat menyebarkan dan menjalankan perangkat lunak apapun, yang dapat mencakup sistem operasi dan aplikasi.

### 2.2.2. OwnCloud

OwnCloud adalah perangkat lunak *open source* yang ditujukan untuk melakukan penyimpanan data di atas *server* yang ditentukan oleh pengguna. Pengguna dapat memilih satu atau lebih folder dalam PC mereka dan melakukan sinkronisasi folder tersebut dengan *server* ownCloud. Sinkronisasi terhadap data lokal dilakukan dengan menyimpan data dalam *local shared directories*, dan data tersebut tersinkronisasi dengan *server*, kemudian tersinkronisasi ke PC lain melalui *desktop client*. *Software* ini dibangun berbasis PHP, MySQL, SQL *Server*, Oracle dan dapat berjalan di atas platform Windows, LINUX, IOS. ownCloud memungkinkan adanya otorisasi penuh dari pengguna atas perlindungan dan pengelolaan data, serta mengintegrasikannya dengan sistem keamanan yang ada. ownCloud menyediakan API bagi pihak ketiga untuk mengembangkan aplikasi yang terintegrasi dengan ownCloud (The ownCloud Developers, 2013).

### 2.2.3. Dropbox

Dropbox adalah sebuah layanan *cloud storage* yang memungkinkan pengguna untuk mengunggah *video*, foto, dokumen dan *file* lainnya untuk *backup storage*. Dropbox memudahkan pengguna untuk berbagi *file* dengan menggunakan sinkronisasi data dan pengguna tidak perlu takut dalam kehilangan atau kerusakan data. Pengaksesan data dapat dilakukan dimana pun baik melalui

komputer maupun *smartphone*. Dropbox memberikan ukuran kapasitas awal yaitu sebesar 2 GB dan menyediakan pilihan untuk memperbesar ruang kapasitas dengan skema pembayaran bulanan. Salah satu kelebihan Dropbox yaitu memberikan keamanan pada data yang disimpan dan data yang terhapus dapat dipulihkan kembali dengan cepat.

#### **2.2.4. Dropbox API**

Dropbox API (*Application Programming Interface*) merupakan sebuah layanan untuk mengintegrasikan aplikasi yang dibangun oleh pihak ketiga dengan aplikasi Dropbox. Dropbox API ini dapat diterapkan ke dalam beberapa bahasa pemrograman diantaranya Python, Ruby, Php, Java dan JavaScript. Aplikasi yang menggunakan Dropbox API dapat dibangun di atas platform Android, iOS, dan *Web*. Hak akses diperlukan oleh Dropbox API untuk melakukan request kepada *server* Dropbox. Hak akses didapat dengan melakukan autentikasi dari aplikasi pihak ketiga ke *server* Dropbox melalui *protocol* OAuth 2. Alasan menggunakan OAuth yaitu salah satunya tidak perlu menyimpan atau mengirimkan *password* Dropbox pengguna. OAuth juga memungkinkan pengguna untuk mengotorisasi hanya hak akses seperangkat terbatas dan pengguna dapat mencabut akses setiap saat.

#### **2.2.5. Sinkronisasi File**

Berdasarkan Vice Versa (2013), sinkronisasi *file* merupakan proses untuk memastikan dua atau lebih lokasi yang memiliki konten *file* yang sama.

Sinkronisasi *file* dapat terjadi satu arah, dimana proses dilakukan dengan menyalin *file* hanya pada satu arah karena kedua lokasi tidak dianggap ekuivalen. Satu lokasi dianggap sebagai sumber dan lokasi lain dianggap sebagai target. Hal ini membuat replika 1:1 dari keseluruhan *file* yang sangat berguna untuk melakukan kebutuhan *backup*, karena hanya *file* baru atau yang berubah saja yang disalin.