



IMPLEMENTASI *OPEN SOURCE* PADA MANAJEMEN *FILE SHARING*

Zainul Arifin¹, Budi Satria Bakti²

Teknik Informatika¹, Universitas Muhammadiyah Jember
Manajemen Informatika², Universitas Muhammadiyah Jember
Email: zainularifin@unmuhjember.ac.id¹, satrio93@unmuhjember.ac.id²

ABSTRAK

File sharing merupakan salah satu bentuk nyata perkembangan era teknologi informasi dalam ruang lingkup penyimpanan data secara digital. Banyaknya macam informasi dan data yang dapat disimpan serta kebutuhan akses terhadap data tersebut mendorong munculnya berbagai metode, diantara berbagai metode tersebut, yang paling umum dan mudah diaplikasikan adalah teknologi *peer-to-peer*. Teknologi *file sharing* telah banyak diimplementasikan di bidang pendidikan, alasan yang mendasari diimplementasikannya teknologi ini adalah data di bidang pendidikan yang semakin meningkat dan bertambah, baik dari kualitas maupun kuantitas, setiap tahunnya. Salah satu permasalahan umum yang sering muncul dari implementasi *file sharing* adalah manajemen *file sharing* dan kemudahan penggunaan *file sharing*, fokus dari penelitian ini adalah mengimplementasikan salah satu perangkat lunak *open source* yaitu *pydio*, yang kemudian digunakan sebagai alat bantu manajemen *file sharing* di lingkungan Universitas Muhammadiyah Jember, sehingga diharapkan manajemen pengaturan dan pengelolaan *file sharing* lebih baik serta mempermudah penggunaan dari sisi *user*.

Kata kunci: *peer-to-peer*, manajemen *file sharing*, *pydio*

ABSTRACT

File sharing is a tangible form of the information technology era development in the digital data storage scope. Many types of information and data that can be stored and the need for accessing to these data have led to the emergence of various methods, among these methods, the most common and easy to apply is *peer-to-peer* technology. *File sharing* technology has been widely implemented in the field of education, the reason for the implementation of this technology is that data in the education sector is increase and build up, both in quality and quantity, every year. One of the common problems that often arise from the implementation of *file sharing* is *file sharing* management and the ease of *file sharing* use, this research focus is implementing one of the open source software, namely *pydio*, which is used as a *file sharing* management tool in the University of Muhammadiyah Jember, so it is hoped that the management of the settings and *file sharing* management will be better and easier to use from the user side.

Keywords: *peer-to-peer*, *file sharing* management, *pydio*

1 PENDAHULUAN

File sharing merupakan salah satu bentuk nyata perkembangan era teknologi informasi dalam ruang lingkup penyimpanan data secara digital. Banyaknya jenis atau macam informasi dan data yang dapat disimpan serta kebutuhan akses terhadap data tersebut mendorong munculnya berbagai metode, seperti metode manual menggunakan media portabel, menggunakan *server* pada suatu jaringan, berbasis internet, atau teknologi *peer-to-peer* (Schollmeier, 2002; Birman, 2005).

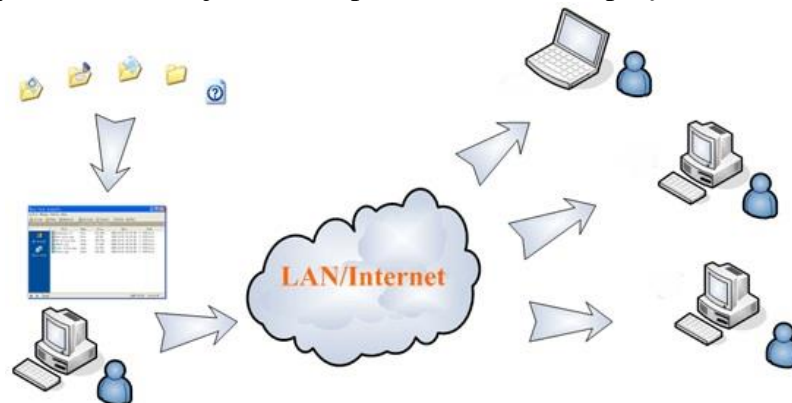
Salah satu permasalahan umum yang sering muncul dari implementasi *file sharing* adalah manajemen *file sharing* dan kemudahan penggunaan *file sharing*. Manajemen *file sharing* terkait dengan tugas utama dari seorang administrator yang meliputi kelancaran dan keamanan sistem yang digunakan, sedangkan kemudahan penggunaan terkait dengan kenyamanan *user* dalam pengoperasian dan penggunaan *file sharing* tersebut. Kasus yang diambil dari penelitian ini adalah mengimplementasikan salah satu perangkat lunak *open source* yaitu *pydio*, yang kemudian digunakan sebagai alat bantu manajemen ` di lingkungan Universitas

Muhammadiyah Jember. Keluaran yang diharapkan adalah dengan digunakannya perangkat lunak ini dapat menjadi solusi dari uraian permasalahan yang ditemukan, yaitu menjadikan manajemen pengaturan dan pengelolaan *file sharing* lebih baik serta mempermudah penggunaan dari sisi *user* dengan penambahan antarmuka melalui perangkat lunak yang digunakan.

2 PENELITIAN TERKAIT

A. *File sharing*

File sharing digunakan untuk berbagi, mencari, dan mengunduh *file* satu ke tempat yang lainnya, atau dapat dikatakan sebagai *transfer* data antar personal komputer atau perangkat sejenis. Teknologi *file sharing* dapat ditemukan pada *Microsoft Windows Networking*, *OS/2*, *Unix/Linux*, dan Keluarga *BSD*. Teknologi ini mulai dikenal sejak tahun 1999 dengan diluncurkannya *Napster* (Menta (2001), Ante (2011)) sebuah program *file sharing* dan *server* direktori yang menghubungkan pemilik dan pencari *file* musik, yang bertujuan untuk meindeks *user* beserta komputernya, sehingga informasi *file* yang kita cari bisa kita ketahui tidak hanya satu lokasi saja akan tetapi semua lokasi tempat *file* tersebut disimpan.



Gambar 1. Ilustrasi *File Sharing*

Teknologi *file sharing* juga tidak mengalami masalah dan kendala dengan adanya perbedaan sistem operasi yang digunakan, karena teknologi ini dapat diaplikasikan antar sistem operasi yang berbeda. Linux merupakan sistem operasi berbasis *open source* menyediakan banyak program yang berfungsi untuk jalur komunikasi antar sistem operasi yang berbeda, misalnya *Microsoft Windows* dan *Linux*.

B. *Peer-to-peer*

Teknologi *peer-to-peer* (P2P) merupakan salah satu teknik yang dipakai dalam *file sharing*, kata “*peer-to-peer*” sendiri berarti “ujung-ke-ujung” yang menganalogikan bahwasannya teknologi ini menghubungkan 2 komputer, akan tetapi sebenarnya *peer-to-peer* adalah suatu model jaringan yang tidak hanya menghubungkan “ujung” satu dengan lainnya, namun tiap komputer yang terhubung tersebut saling berhubungan secara dinamis dan secara aktif terlibat dalam proses transportasi komunikasi data dan pembagian *bandwidth*, dimana pada umumnya proses tersebut merupakan tanggung jawab *server* pusat (Kaciak, 2013).

Karakteristik utama jaringan ini adalah tidak terdapat sebuah *server* pusat yang mengatur *clien*, karena memang setiap komputer bertindak sebagai *server* untuk

komputer klien lainnya atau dengan kata lain bahwa *host* dapat dijadikan *server* dan juga menjadi *client* secara bersamaan.



Gambar 2. Arsitektur *Peer-to-Peer*

Adapun beberapa kelebihan jaringan *peer-to-peer* adalah antar komputer dalam jaringan dapat saling berbagi-pakai fasilitas yang dimilikinya seperti: *harddisk*, *fax/modem*, dan *printer*, implementasi, operasional, dan perawatannya murah dan mudah, karena tidak memerlukan adanya *server*, sistem dan aktivitas tidak tergantung pada satu *server*. Sedangkan kekurangan dari jaringan *peer-to-peer* adalah *troubleshooting* jaringan relatif lebih sulit, tingkat keamanannya rendah karena sistem keamanan jaringan ditentukan oleh masing – masing *user* dengan mengatur fasilitas yang dimiliki, semakin banyak yang di-*share*, akan mempengaruhi kinerja komputer yang dipakai, terutama *bandwidth* (Suteni, 2012).

C. *Pydio*

Pydio merupakan perangkat lunak *file server* berbasis *open source* yang dapat diakses menggunakan komputer, maupun *smartphone*. *Pydio* merupakan *file manager* berbasis *AJAX* dengan fitur yang banyak dan *interface* menyerupai *system explorer* standar pada setiap sistem operasi. *Pydio* memiliki kemampuan untuk melakukan *streaming* video dan audio secara *online*, menampilkan gambar maupun *text*, dan dapat melakukan *sharing* dokumen – dokumen, *Pydio* memiliki *admin control panel* untuk manajemen *user* (De Jeu, 2007).

D. Studi Hasil Penelitian Sebelumnya

Distributed Files Sharing Management: A File Sharing Application Using Distributed Computing Concepts, penelitian ini mencoba memberikan solusi untuk permasalahan *file sharing* pada kasus terjadinya pengendali utama (*controller*) yang mati secara tiba – tiba, konsep yang diterapkan adalah sistem terdistribusi yang diterapkan dengan membangun pengendali replika (*replica controller*) yang berada di lokasi berbeda.

Hasil akhir yang diperoleh, sistem yang dibangun dengan 5 buah *node* yaitu : *controller*, *replica controller*, *database server* dan 2 *clients* mampu berjalan dengan sukses, terbukti ketika *controller* di kondisikan mati tiba – tiba maka *replica controller* secara otomatis bekerja, dan *transfer* data yang dilakukan tidak mengalami *packet loss* (Malgaonkar, Surve, & Hirave, 2012).

E. Metode Penelitian

Tahapan penelitian yang dilakukan terbagi dalam 4 tahap yaitu persiapan sistem, pengaturan sistem, uji coba sistem dan pembahasan, kemudian tahap

terakhir adalah penyusunan laporan. Deskripsi tahapan penelitian bisa dilihat pada gambar 3 di bawah ini, yang kemudian diikuti dengan penjelasan mengenai masing – masing tahapan penelitian.



Gambar 2. Alur Penelitian

Tahap persiapan sistem adalah melakukan instalasi *pydio* pada *server*, tahap ini juga sekaligus merupakan analisa terhadap kebutuhan lingkungan perangkat lunak yang digunakan. Setelah dilakukan instalasi maka tahap selanjutnya adalah pengaturan sistem, yang meliputi pembuatan *group user* serta penentuan hak akses dari masing – masing *group user* tersebut. Tahap setelah pengaturan sistem adalah uji coba dan pembahasan, tahap ini dilakukan untuk mengetahui sejauh mana kinerja dari *pydio* untuk alat bantu manajemen *file sharing*, selain itu juga dilakukan evaluasi terhadap penggunaan dari sisi *user*. Tahapan terakhir yang dilakukan adalah penyusunan dokumen dalam rangka pendokumentasian seluruh kegiatan penelitian ini.

3. HASIL DAN PEMBAHASAN

A. Tahap persiapan

Tahap ini merupakan tahap awal untuk melakukan implementasi *pydio* sebagai alat bantu manajemen *file sharing* dimana dilakukan proses instalasi pada *server* yang digunakan. Pada penelitian ini digunakan sistem operasi Ubuntu 14.10 *server*. Konfigurasi *hostname server1.example.com* dan konfigurasi *IP Address* 192.168.0.100. rincian proses pada tahap persiapan dapat dilihat pada tabel di bawah ini.

Tabel 31. Proses Tahap Persiapan

NO	PROSES	INSTRUKSI
1.	Instalasi <i>pydio</i> dari repository	<code>apt-get install pydio</code>
2.	Salin file konfigurasi pada folder konfigurasi web server	<code>cp /usr/share/doc/pydio/apache2.sample.conf /etc/apache2/sites-enabled/pydio.conf</code>
3.	Melakukan restart web server	<code>service apache2 restart</code>
4.	Instalasi basis data	<code>apt-get install mariadb-server php5-mysql</code>
5.	Masuk ke sistem basis data	<code>mysql -u root -p</code>
6.	Membuat basis data dengan nama “ <i>pydiodb</i> ”	<code>create database pydiodb;</code>

7.	Membuat username “pydiouser” kemudian menentukan passwordnya dengan “pydiopassword”	create user pydiouser@localhost identified by 'pydiopassword';
8.	Menentukan hak akses dari username tersebut	grant all on pydiodb.* to pydiouser@localhost;
9.	Menginstruksikan sistem basis data untuk mengatur ulang tabel hak akses	flush privileges;
10.	Keluar dari sistem basis data	Exit
11.	Mengatur mode untuk mcrypt	php5enmod mcrypt
12.	Melakukan restart service web server	service apache2 restart
13.	Melakukan restart service sistem basis data	service mysql restart

Setelah melakukan urutan keseluruhan proses di atas, maka tahap persiapan telah selesai dilakukan, sehingga kita siap untuk melakukan tahapan selanjutnya.

B. Tahap pengaturan sistem

Tahap ini meliputi pembuatan *group user* serta penentuan hak akses dari masing - masing *group user* tersebut. *Group user* nantinya akan dibagi berdasarkan *level user* dan akan diberikan hak akses yang berbeda – beda untuk masing – masing *level user*.

1. Perancangan *group user*

Perancangan *level user* bertujuan memberi hak akses yang berbeda – beda untuk masing – masing *user*. Berikut ini adalah hasil dari perancangan *level user* yang telah dilakukan.

Tabel 2. *Level User*

Folder	User : umum	User : dosen	User : admin
Ebook	Read	Read	Read, write
Games	Read	Read	Read, write
Kuliah	Read	Read, write	Read, write
Manga	Read	Read	Read, write
Master	Read	Read	Read, write
Movies	Read	Read	Read, write
Musics	Read	Read	Read, write
Photos	Read	Read	Read, write
Upload	Read, write	Read, write	Read, write

Keterangan:

- Kolom *folder* adalah kategori data yang telah disediakan.
- Keterangan “*Read*” menjelaskan bahwa *user* yang bersangkutan dapat melakukan akses membaca atau melihat isi dari *folder* di kolom kiri.
- Keterangan “*Write*” menjelaskan bahwa *user* yang bersangkutan dapat melakukan akses menulis/mengisi *file* pada *folder* di kolom kiri.

2. Perancangan kelompok data

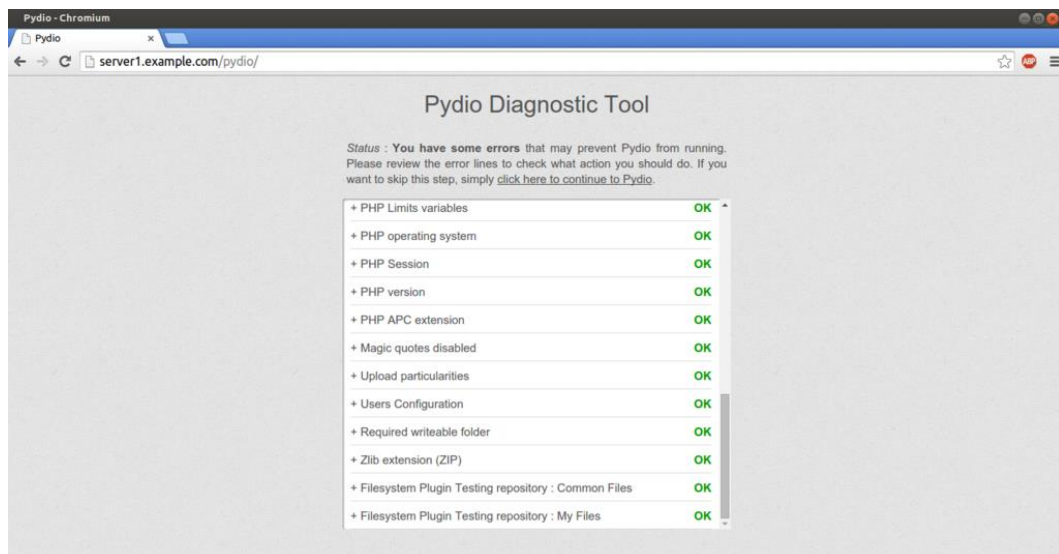
Perancangan kelompok data dilakukan untuk mengorganisasi data dan informasi. Berdasarkan pada jenis data yang tersedia maka pengelompokan data dibagi menjadi 4 kategori utama sebagai berikut:

- Data teks
 - *File office* (.doc, .xls, .ppt)
- Data gambar
 - Foto – foto
 - *Scan*
- Data audio
 - Lagu
 - Musik
- Data video
 - Film
 - Video klip

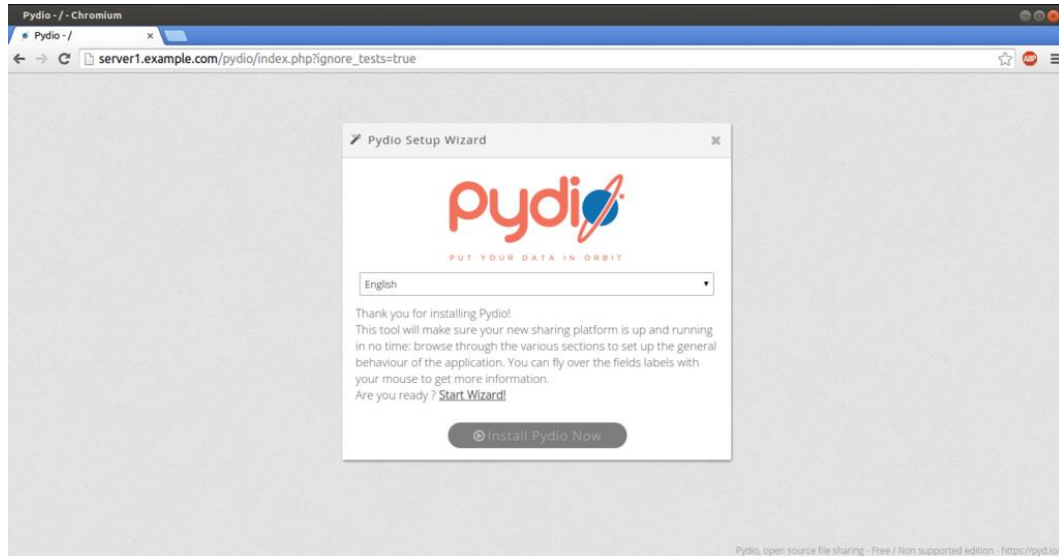
Setelah dikategorikan menjadi 4 kategori utama kemudian dapat dilakukan penentuan sub kategori untuk masing – masing kategori data yang bersesuaian.

3. Implementasi dan uji coba

Implementasi dan uji coba perangkat lunak *pydio* hanya dilakukan meliputi konfigurasi antarmuka serta melakukan *login* untuk *username* admin. Beberapa dokumentasi dari hasil implementasi dan uji coba bisa dilihat sebagai berikut:

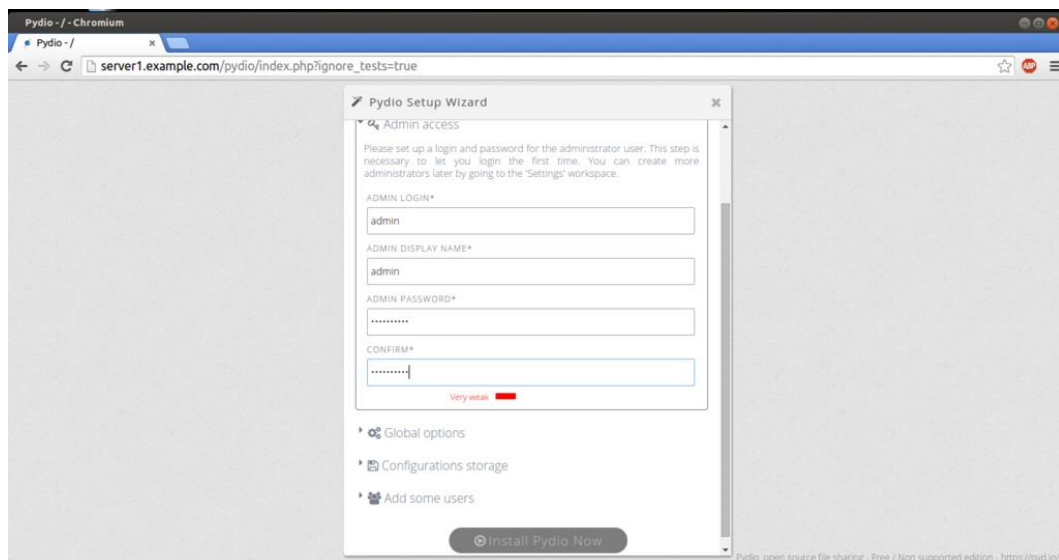
Gambar 3. *Run Pydio*

Gambar 4 adalah tampilan ketika kita mengakses *pydio* melalui alamat *hostname/pydio* untuk melakukan konfigurasi web yang berfungsi sebagai tampilan nantinya. Konfigurasi dilakukan dengan pengaturan otomatis menggunakan *wizard* seperti yang ditunjukkan pada gambar 5 di bawah ini:



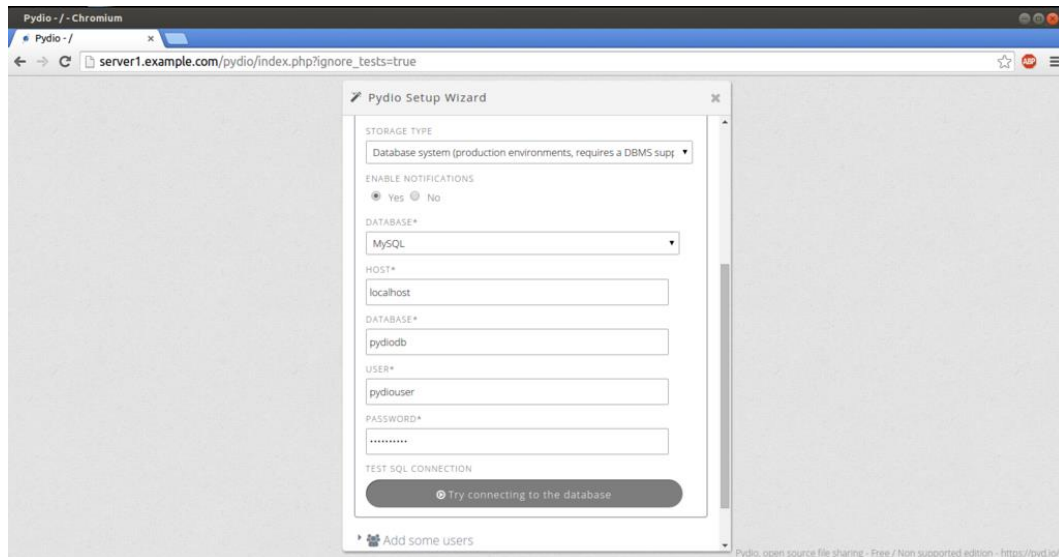
Gambar 5. *Pydio Wizard*

Setelah masuk pada *wizard* tersebut maka kita lakukan instalasi. Kita akan diminta untuk mendaftarkan *username* dan *password* agar dapat *login* dan melakukan kegiatan *browsing*, *upload*, dan *download*.



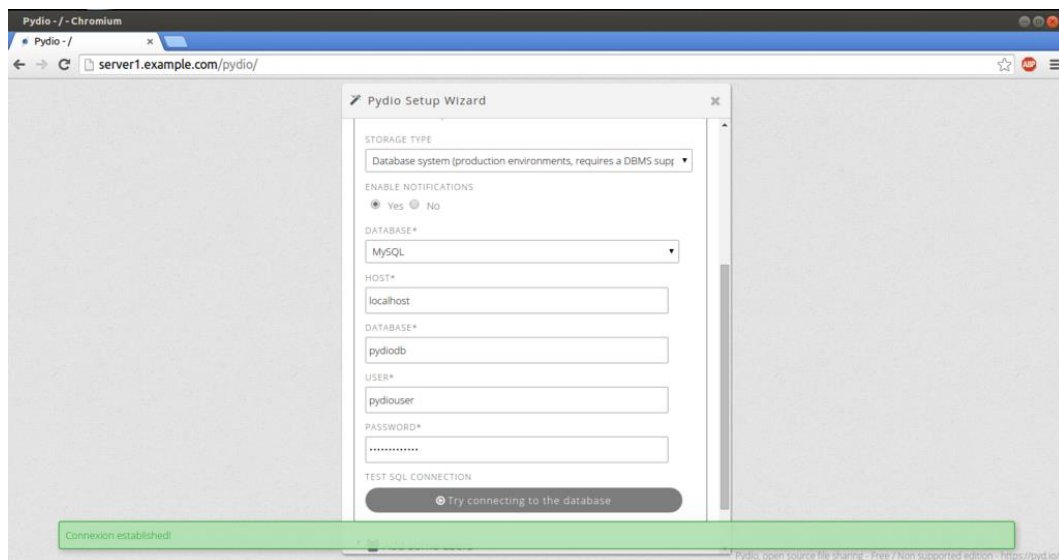
Gambar 6. Form Isian *Username* dan *Password*

Pada proses ini dilakukan pengisian *user* = *admin* dan *password* = *howtoforge*. Kemudian dilakukan pengaturan untuk koneksi antara *pydio* dengan sistem basis data yang digunakan seperti yang ditunjukkan pada gambar 6.



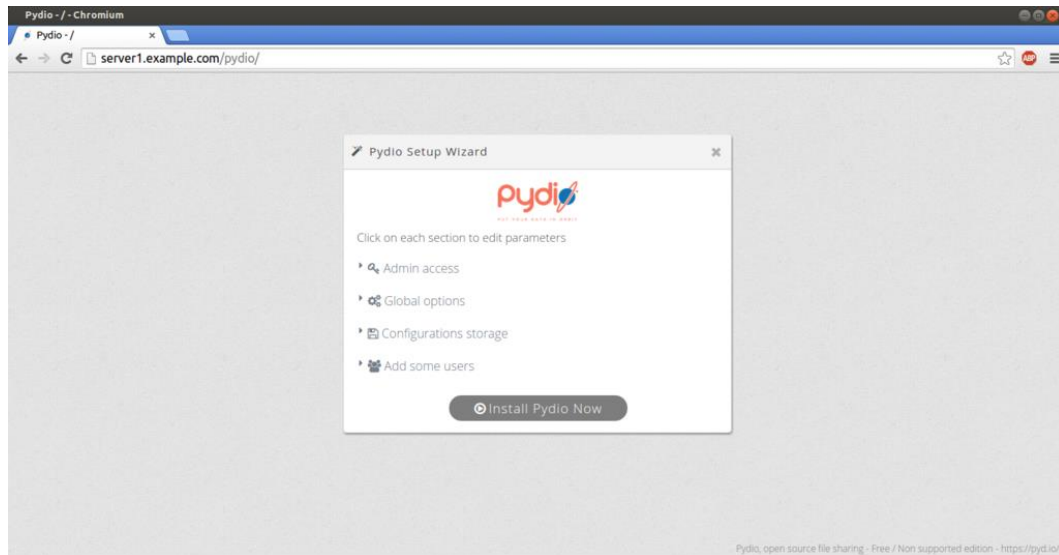
Gambar 7. Form Isian Sistem Basis Data

Untuk pengaturan seperti di atas, isian harus disesuaikan dengan apa yang sudah dilakukan seperti pada tabel 7, sehingga untuk basis data kita isikan *pydiodb*, *username* kita isikan *pydiouser*, dan *password* kita isikan *pydiopassword*. Setelah semua isian lengkap dan benar kita lakukan koneksi dengan basis data seperti yang ditunjukkan oleh gambar 8 di bawah ini.



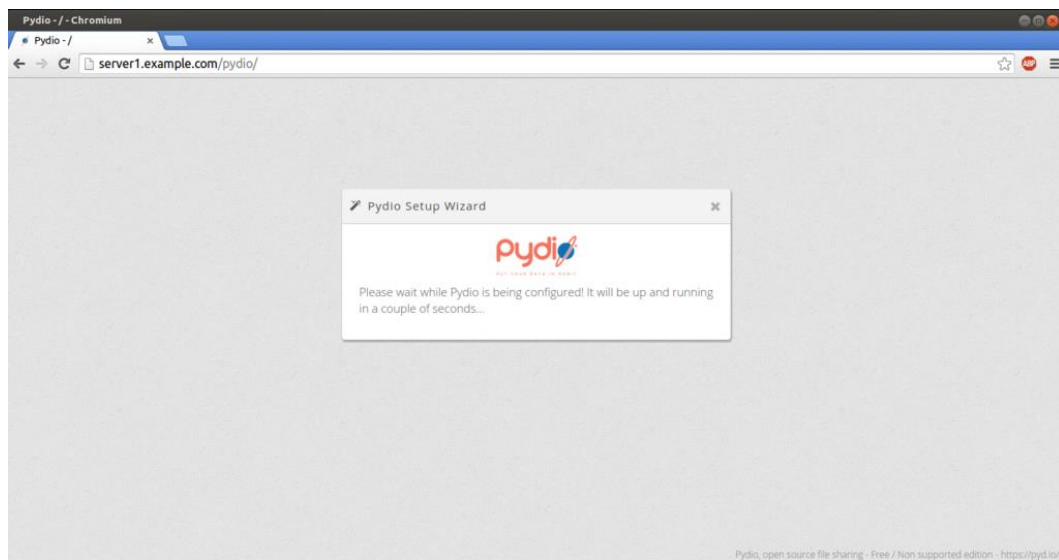
Gambar 8. Koneksi dengan Basis Data

Apabila muncul seperti gambar 8 di atas maka koneksi basis data dengan *pydio* telah berhasil. Selanjutnya akan keluar tampilan halaman awal seperti pada gambar 9 dari *pydio* yang telah kita konfigurasi.



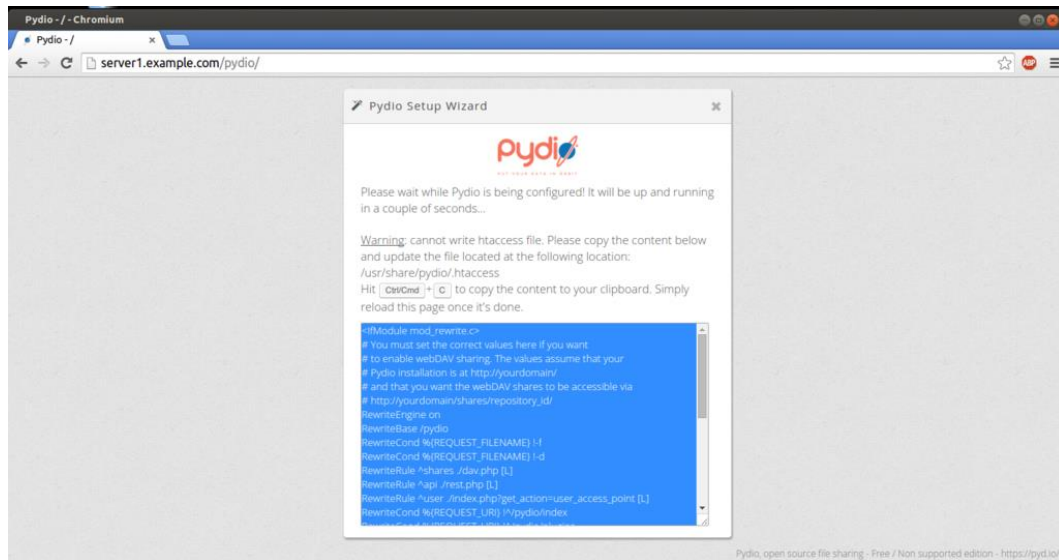
Gambar 39. Halaman Awal

Pada bagian ini kita dapat juga merubah kembali konfigurasi yang telah dilakukan, menambah konfigurasi, atau dapat juga menambahkan *username* untuk *login*. Apabila semua telah dirasa cukup maka proses dapat kita lanjutkan dengan menekan tombol “*Install Pydio Now*”.



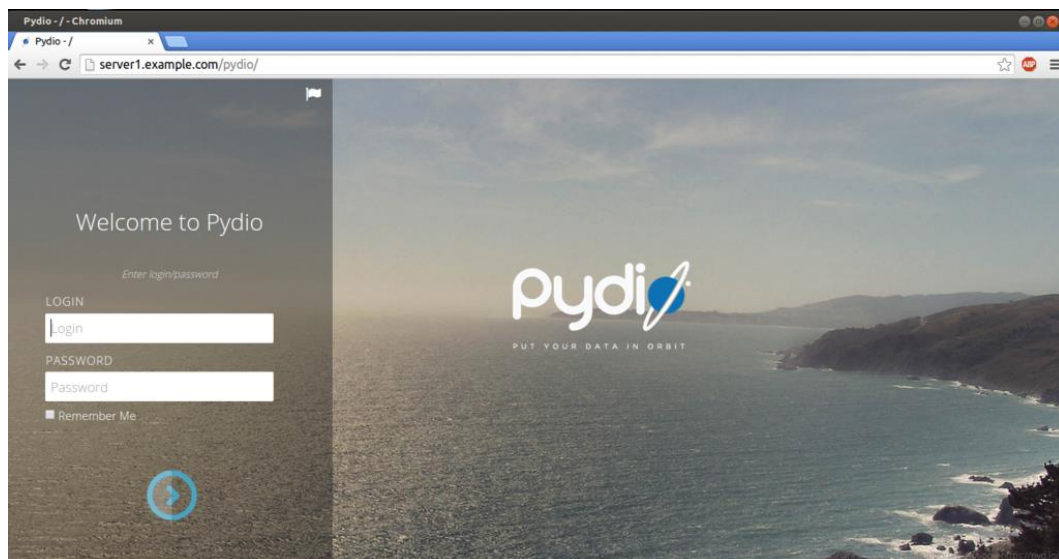
Gambar 10. Proses Setup

Gambar 10 di atas merupakan tampilan ketika dilakukan setup atau instalasi, waktu yang dibutuhkan tergantung dari perangkat keras yang digunakan. Setelah *setup* selesai maka akan muncul tampilan seperti gambar 11.



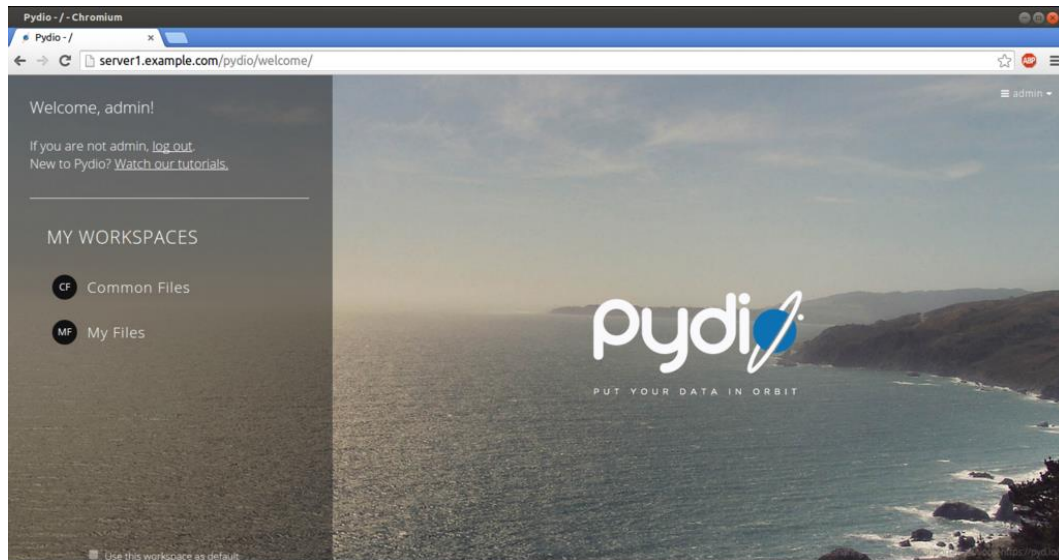
Gambar 11. Halaman Notifikasi Setup Selesai

Apabila muncul tampilan seperti gambar di atas maka *setup* telah selesai dan kita lakukan penyalinan *file .htaccess* ke dalam *folder/usr/share/pydio/.htaccess*:



Gambar 12. Halaman Awal Pydio

Gambar 12 menunjukkan tampilan halaman awal *pydio* yang siap kita pakai. Untuk *login* maka digunakan *username* dan *password* yang telah dibuat.

Gambar 13. Halaman *Login Administrator*

Gambar di atas menunjukkan bahwa kita telah berhasil *login* sebagai administrator. Sampai dengan proses ini maka *pydio* siap digunakan sebagai media untuk pengaturan atau manajemen *file sharing* yang mencakup *upload*, *download*, dan pengaturan hak akses bagi *user* yang nantinya menggunakan aplikasi ini.

Kesimpulan yang diperoleh sampai dengan laporan kemajuan ini dikerjakan adalah: Persiapan manajemen *file sharing* telah dilakukan, mencakup persiapan *server*, konfigurasi *server*, dan konfigurasi *pydio*. Serta *Pydio* telah berhasil berjalan dengan baik di *server* yang telah dikonfigurasi.

Adapun saran yang terkait dengan penelitian ini adalah: Perlu dilakukan pengujian terhadap kemampuan sistem yang telah dirancang dan dibangun.

4. DAFTAR PUSTAKA

1. Ante, Spencer. 2011. *Inside Napster: Business Week*. Tersedia di http://www.businessweek.com/2000/00_33/b3694001.htm diakses 24 April 2014.
2. Birman, K. P. 2005. *Reliable Distributed Systems*.
3. De Jeu, C. 2007. *About Pydio*. Tersedia di <http://pyd.io/about/> diakses 26 April 2014
4. Kaciak, G., R. 2013. *Pengertian Jaringan peer-to-peer*. Tersedia di <http://dosen.gufron.com/artikel/pengertian-jaringan-peer-to-peer-p2p/7/> diakses 25 April 2014.
5. Malgaonkar, S., Surve S., & Hirave, T. 2012. *Distributed Files Sharing Management: A File Sharing Application Using Distributed Computing Concepts*. *IEEE International Conference on Computational Intelligence and Computing Research*.
6. Menta, Richard. 2001. *Napster Clones Crush Napster. Take 6 out of the Top 10 Downloads on CNet. MP3 Newswire*. <http://www.mp3newswire.net/stories/2001/topclones.html> diakses 24 April 2014.

7. Schollmeier, R. 2002. *A Definition of Peer-to-Peer Networking for the Classification of Peer-to-Peer Architectures and Applications. Proceedings of the First International Conference on Peer-to-Peer Computing IEEE*, pp. 149-160.
8. Suteni. 2012. *Pengertian jaringan peer-to-peer*. Tersedia di <http://suteni.blogspot.com/2012/05/pengertian-jaringan-peer-to-peer.html>