

# PEMANFAATAN *OPEN SOURCE FILE SHARING* MENGUNAKAN *NEXTCLOUD* DALAM BERTUKAR DATA PADA DINAS KEARSIPAN DAN PERPUSTAKAAN KOTA MATARAM

*(Utilization of Open Source File Sharing Using Nextcloud for Data Exchange at the Archival and Library Office of Mataram City)*

Muhammad Eysar Assazily<sup>[1]</sup>, Mohammad Zaenuddin Hamidi<sup>[1]</sup>, Firmanda Rizky Arinanta<sup>[2]</sup>, Pahrul Irfan<sup>[1]</sup>

<sup>[1]</sup>Department of Informatics Engineering, Mataram University

Jl. Majapahit 62, Mataram, Lombok NTB, INDONESIA

<sup>[2]</sup>Department of Archives and Library of Mataram City

Jl. Dr. Sudjono, Lingkar Selatan, Mataram, Lombok NTB, INDONESIA

Email: [fld022070@student.unram.ac.id](mailto:fld022070@student.unram.ac.id), [zaen\\_hamidi@unram.ac.id](mailto:zaen_hamidi@unram.ac.id), [firmandarizkyarinanta@outlook.com](mailto:firmandarizkyarinanta@outlook.com),  
[irfan@staff.unram.ac.id](mailto:irfan@staff.unram.ac.id)

## Abstrak

*Pengelolaan arsip digital di instansi pemerintahan sering menghadapi kendala dalam efisiensi, keamanan, dan keteraturan data. Kegiatan ini bertujuan untuk mengoptimalkan proses pertukaran dan pengelolaan arsip digital di Dinas Kearsipan dan Perpustakaan Kota Mataram melalui pemanfaatan platform open source Nextcloud. Sistem ini diterapkan sebagai solusi file sharing terpusat bernama Kotak Diarpus yang mendukung kolaborasi dan keamanan data antarpegawai. Untuk meningkatkan efisiensi operasional, dikembangkan serangkaian skrip otomatisasi berbasis Python yang diintegrasikan dengan Task Scheduler guna menjalankan proses kompresi, ekstraksi, pencadangan, dan pemantauan aktivitas file secara terjadwal tanpa intervensi manual. Implementasi sistem otomatisasi ini diuji melalui konfigurasi dan pengujian langsung di lingkungan instansi. Hasil menunjukkan bahwa seluruh fungsi berjalan stabil dan efektif dalam menjaga kontinuitas data. Penerapan Nextcloud dan otomatisasi berbasis Python terbukti mampu meningkatkan efisiensi, keamanan, dan keberlanjutan pengelolaan arsip digital, sekaligus meminimalkan risiko kehilangan data. Hasil kegiatan ini diharapkan menjadi dasar pengembangan sistem digital terintegrasi yang lebih andal pada instansi pemerintah.*

**Keywords:** Nextcloud, Otomatisasi, File Sharing, Python, Task Scheduler, Arsip Digital.

## 1. PENDAHULUAN

Perkembangan teknologi informasi telah mendorong berbagai lembaga, termasuk instansi pemerintahan, untuk beralih ke sistem digital dalam mendukung administrasi dan pengelolaan data. *Cloud computing* menjadi salah satu solusi yang banyak digunakan karena memungkinkan proses penyimpanan, pengelolaan, dan kolaborasi data dilakukan secara daring. Bagi lembaga arsip dan perpustakaan yang memiliki volume dokumen tinggi, sistem penyimpanan berbasis *cloud* yang aman, fleksibel, dan terpusat menjadi kebutuhan penting untuk menjaga keandalan serta efisiensi pengelolaan data.

Salah satu platform *open source* yang banyak digunakan untuk berbagi *file* dan kolaborasi daring adalah *Nextcloud*. Sebagai platform terbuka, *Nextcloud* memungkinkan pengguna mengelola data secara mandiri melalui server lokal maupun internet, dengan kendali penuh terhadap keamanan dan privasi. Selain itu, *Nextcloud* mendukung kolaborasi dalam bentuk pengeditan dokumen bersama dan penjadwalan aktivitas [1], [2]. Menurut Gufron dan Sri Syahriarti [3], penerapan *Nextcloud* dapat meningkatkan efisiensi pengelolaan data serta mengurangi ketergantungan terhadap penyimpanan fisik yang rentan terhadap kerusakan.

Dinas Kearsipan dan Perpustakaan Kota Mataram membutuhkan sistem pengelolaan dan keamanan data digital yang optimal. Hal ini disebabkan oleh banyaknya dokumen penting yang harus diarsipkan dan dibagikan antarbidang sehingga diperlukan sistem yang mampu menjamin keutuhan, ketersediaan, dan kerahasiaan data. Selain itu, proses pengarsipan manual masih berpotensi menimbulkan risiko kehilangan data akibat kesalahan pengguna, kerusakan media penyimpanan, maupun keterbatasan kapasitas penyimpanan konvensional. Oleh karena itu, pemanfaatan *Nextcloud* menjadi salah satu solusi yang dapat mendukung peningkatan efisiensi kerja dan menjaga keutuhan arsip digital di lingkungan instansi.

Salah satu fokus kegiatan diarahkan pada penerapan otomatisasi terhadap berbagai tugas rutin yang mendukung pertukaran dan pengelolaan data secara lebih terstruktur dan berkelanjutan. Penerapan otomatisasi ini penting karena banyak aktivitas pengelolaan data yang bersifat repetitif, seperti pencadangan (*backup*), pemindahan *file*, dan pemantauan aktivitas pengguna, yang jika dilakukan secara manual dapat menyita waktu dan meningkatkan potensi kesalahan manusia. Dengan adanya otomatisasi, tugas-tugas tersebut dapat dijalankan secara terjadwal dan konsisten tanpa keterlibatan langsung pengguna, sehingga sumber daya manusia dapat lebih difokuskan pada kegiatan strategis lainnya.

Penerapan otomatisasi dalam pengelolaan data digital dapat diwujudkan melalui penggunaan bahasa pemrograman *Python* yang dikenal fleksibel, mudah digunakan, dan memiliki pustaka standar yang luas untuk mengelola *file* serta proses sistem secara efisien [4]. *Python* juga dapat diintegrasikan dengan layanan penjadwalan seperti *Task Scheduler* untuk menjalankan skrip secara otomatis pada waktu tertentu, sehingga berbagai tugas rutin dapat berjalan tanpa keterlibatan langsung pengguna. Berdasarkan Al-Hadi et al. [5], integrasi penjadwalan otomatis mampu meningkatkan efisiensi proses dan mengurangi keterlambatan eksekusi tugas secara signifikan. Oleh karena itu, kegiatan pengabdian ini berfokus pada pemanfaatan *Nextcloud* yang dikombinasikan dengan otomatisasi berbasis *Python* dan *Task Scheduler* untuk mendukung proses pertukaran data di Dinas Kearsipan dan Perpustakaan Kota Mataram.

## 2. TINJAUAN PUSTAKA

### 2.1. Cloud Computing

*Cloud computing* atau komputasi awan merupakan paradigma komputasi modern yang menyediakan sumber daya seperti penyimpanan, server, dan aplikasi melalui internet secara fleksibel sesuai kebutuhan pengguna. Teknologi ini menggabungkan konsep virtualisasi dan arsitektur berorientasi layanan, sehingga pengguna dapat mengakses layanan komputasi tanpa harus memikirkan detail implementasi maupun pengelolaannya. Dengan berbagai model seperti *SaaS*, *PaaS*, dan *IaaS*, serta tipe penerapan *public*, *private*, *community*, dan *hybrid cloud*, teknologi ini mampu menghadirkan solusi yang efisien, terukur, aman, serta hemat biaya bagi organisasi [6].

Pemanfaatan *cloud computing* telah memberikan dampak signifikan dalam berbagai sektor, termasuk pendidikan, pemerintahan, dan bisnis. Teknologi ini memungkinkan pengelolaan data secara terpusat, akses jarak jauh, serta otomatisasi proses kerja yang meningkatkan efisiensi dan produktivitas. Menurut Rumetna [7], *cloud computing* mampu menekan biaya investasi infrastruktur TI, meningkatkan keandalan sistem, dan mempercepat kolaborasi antarpengguna. Selain itu, kemampuannya dalam menyesuaikan kapasitas penyimpanan secara dinamis menjadikannya solusi strategis dalam menghadapi pertumbuhan data dan kebutuhan layanan digital yang terus meningkat, baik pada organisasi bisnis maupun lembaga publik seperti perpustakaan dan arsip.

### 2.2. Nextcloud

*Nextcloud* merupakan platform *cloud storage* berbasis *open source* yang memungkinkan penyimpanan, sinkronisasi, dan kolaborasi data secara terpusat melalui server lokal maupun internet. Sistem ini menyediakan akses fleksibel dari berbagai perangkat, baik secara *offline* melalui jaringan lokal maupun *online* melalui jaringan publik, serta dilengkapi dengan fitur keamanan seperti autentikasi dan enkripsi. *Nextcloud* juga dapat diintegrasikan dengan layanan seperti *Apache Web Server* dan *MariaDB* untuk mendukung kinerja server yang stabil dan andal [8].

Pemanfaatan *Nextcloud* memberikan kontribusi signifikan terhadap efisiensi, keamanan, dan fleksibilitas pengelolaan data. Berdasarkan hasil kajian Albert dkk. [9], penggunaan *Nextcloud* dalam lingkungan organisasi dan perusahaan mampu meningkatkan produktivitas melalui kemudahan kolaborasi dan akses data dari mana saja. Platform ini mendukung distribusi data secara aman menggunakan koneksi jaringan publik, memungkinkan kerja tim jarak jauh, serta memberikan kontrol penuh terhadap hak akses pengguna dan privasi data. Selain itu, integrasinya dengan berbagai aplikasi pihak ketiga menjadikan *Nextcloud* solusi yang adaptif terhadap kebutuhan institusi, baik di sektor pendidikan, bisnis, maupun pemerintahan, dalam menciptakan sistem manajemen arsip digital yang efisien dan terstruktur.

### 2.3. Otomatisasi

Otomatisasi merupakan proses penerapan teknologi untuk menjalankan prosedur atau pekerjaan secara otomatis tanpa keterlibatan manusia secara langsung. Tujuan utamanya adalah meningkatkan efisiensi, efektivitas, ketepatan, serta konsistensi dalam pelaksanaan tugas. Sistem otomatisasi mampu mengurangi kesalahan manusia (*human error*), mempercepat waktu penyelesaian pekerjaan, dan memungkinkan pengolahan data dalam jumlah besar secara lebih terstruktur dan akurat [10].

Pemanfaatan otomatisasi telah terbukti memberikan dampak positif dalam meningkatkan produktivitas dan efektivitas kerja di berbagai sektor. Temuan dari Jambak dkk. [11] menunjukkan bahwa penerapan sistem otomatisasi perkantoran di Badan Wakaf Al-Qur'an berhasil mengubah proses kerja manual menjadi sistem digital yang serba otomatis, sehingga komunikasi internal dan distribusi informasi antarbagian menjadi lebih lancar.

Teknologi otomatisasi tidak hanya membantu mempercepat penyelesaian tugas administratif, tetapi juga berperan dalam pengambilan keputusan yang lebih tepat melalui penyediaan data yang *real-time* dan terintegrasi.

#### 2.4. Python

*Python* merupakan bahasa pemrograman tingkat tinggi yang dikenal memiliki *syntax* sederhana, fleksibilitas tinggi, dan komunitas pengguna yang luas. Salah satu keunggulan utama *Python* adalah sifatnya yang *dynamically typed*, di mana tipe data dikenali secara otomatis tanpa perlu dideklarasikan secara eksplisit, sehingga mempermudah proses pengembangan, terutama bagi pemula. *Python* juga mendukung berbagai paradigma pemrograman seperti prosedural, berorientasi objek, dan fungsional, menjadikannya pilihan yang cocok untuk berbagai jenis proyek, mulai dari aplikasi berskala kecil hingga sistem kompleks [4].

Pemanfaatan *Python* tidak hanya terbatas pada pengembangan aplikasi umum, tetapi juga mencakup bidang-bidang seperti automasi, analisis data, kecerdasan buatan, dan *Internet of Things (IoT)*. Berdasarkan Semendawai dkk. [12], *Python* dapat digunakan untuk merancang skrip otomatisasi dalam memantau pemakaian data harian pada kartu IoT secara efisien tanpa perlu intervensi manual. Melalui kemampuannya dalam memproses data secara cepat dan mengotomatisasi tugas-tugas rutin, *Python* mampu meningkatkan efisiensi operasional dan akurasi sistem monitoring. Kemudahan integrasi dengan basis data dan layanan penjadwalan juga menjadikan *Python* sebagai alat yang efektif untuk mendukung implementasi sistem otomatisasi di berbagai lingkungan kerja, termasuk lembaga pendidikan, bisnis, dan pemerintahan.

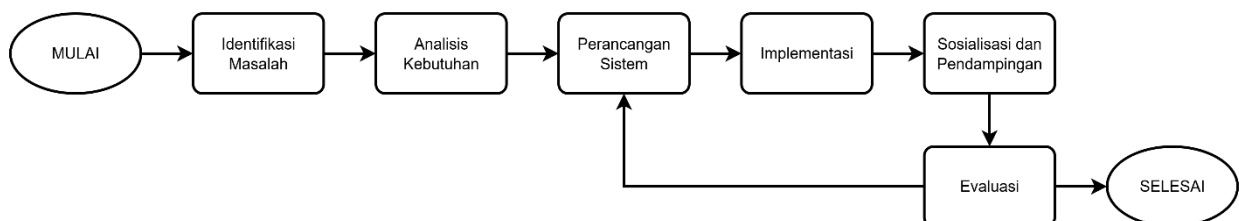
#### 2.5. Task Scheduler

*Task Scheduler* merupakan komponen penting dalam sistem operasi yang berfungsi untuk menjadwalkan eksekusi program, perintah, atau skrip secara otomatis sesuai dengan waktu yang telah ditentukan atau pada interval tertentu. Melalui *Task Scheduler*, pengguna dapat mengatur berbagai aktivitas seperti menjalankan aplikasi, mengirimkan notifikasi, mematikan komputer, hingga menjalankan skrip secara berkala tanpa keterlibatan langsung pengguna [5]. Fitur ini sangat bermanfaat dalam meningkatkan efisiensi kerja sistem, karena memastikan tugas-tugas rutin dapat berjalan tepat waktu secara konsisten. Selain itu, *Task Scheduler* juga mendukung automasi pada proses pembaruan aplikasi dan pengelolaan sistem, sehingga membantu pengguna dalam meminimalkan kesalahan manusia dan mempercepat proses kerja.

Pemanfaatan *Task Scheduler* terbukti efektif dalam mendukung efisiensi administrasi dan operasional di berbagai lingkungan kerja. Menurut Setyawati dkk. [13], penerapan *Task Scheduler* dalam sistem pelaporan kegiatan kemahasiswaan di Fakultas Teknik Universitas Negeri Malang mampu mengoptimalkan penyusunan laporan pertanggungjawaban keuangan (LPJ) melalui penjadwalan otomatis. Fitur pengingat (*reminder*) dan notifikasi otomatis memungkinkan mahasiswa menyelesaikan laporan tepat waktu, sementara pihak keuangan dapat memantau proses pelaporan secara *real-time* tanpa tatap muka langsung. Hal ini menunjukkan bahwa *Task Scheduler* tidak hanya berperan dalam automasi teknis, tetapi juga dalam meningkatkan ketertiban, akuntabilitas, dan efektivitas proses administrasi di lingkungan pendidikan maupun organisasi.

### 3. METODE PENGABDIAN MASYARAKAT

Berikut merupakan metode yang digunakan dalam menganalisis dan menerapkan pemanfaatan platform *open source file sharing Nextcloud* pada Dinas Kearsipan dan Perpustakaan Kota Mataram, yang dikombinasikan dengan penggunaan skrip otomatisasi berbasis *Python* dan *Task Scheduler* untuk mendukung efisiensi proses pertukaran dan pengelolaan data digital:



Gambar 3.1 Flowchart Metode Pengabdian Masyarakat

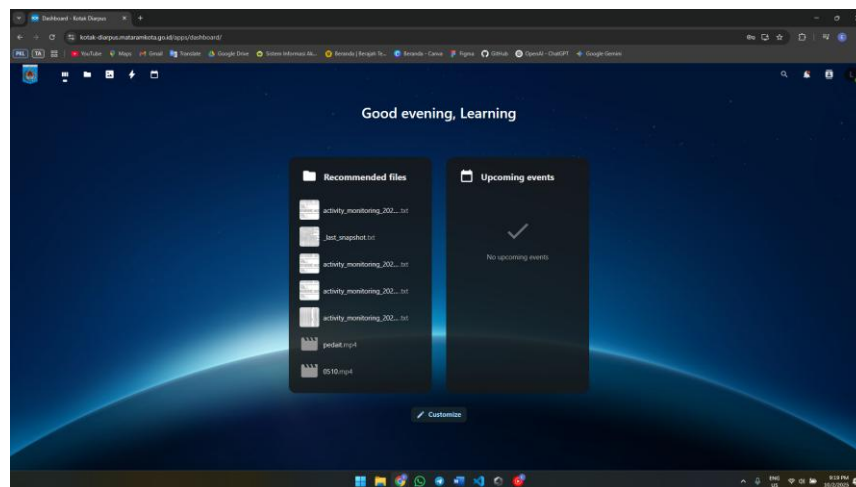
- Identifikasi Masalah: Tahapan ini bertujuan untuk mengenali permasalahan yang dihadapi dalam pengelolaan dan pertukaran arsip digital di Dinas Kearsipan dan Perpustakaan Kota Mataram. Melalui proses observasi, ditemukan beberapa kendala seperti belum optimalnya pemanfaatan fitur *Nextcloud* untuk mendukung efisiensi kerja pegawai, serta beberapa proses pertukaran data yang masih dilakukan secara manual.
- Analisis Kebutuhan: Pada tahap ini dilakukan analisis terhadap kebutuhan sistem dan pengguna dalam pemanfaatan *Nextcloud* agar dapat mendukung kegiatan berbagi dan pengelolaan data secara lebih efisien.

- Analisis mencakup identifikasi kebutuhan perangkat pendukung, pemetaan fitur *Nextcloud* yang relevan, serta perancangan mekanisme otomatisasi yang dapat membantu proses pertukaran data.
- Perancangan Sistem: Tahap ini mencakup penyusunan rancangan pengembangan dan pemanfaatan sistem berbasis *Nextcloud* yang terintegrasi dengan mekanisme otomatisasi. Perancangan diarahkan untuk meningkatkan efisiensi kerja dan menjaga keamanan data, dengan memperhatikan kesesuaian antara kebutuhan pengguna, kemampuan sistem, dan sumber daya yang tersedia.
  - Implementasi: Tahapan implementasi dilakukan dengan menerapkan hasil rancangan ke dalam sistem *Nextcloud*. Proses ini meliputi pengujian fungsi dan penerapan integrasi terhadap mekanisme otomatisasi yang telah dirancang sebelumnya. Tujuannya adalah memastikan seluruh proses berjalan stabil, efisien, dan sesuai dengan kebutuhan.
  - Sosialisasi dan Pendampingan: Tahap sosialisasi dan pendampingan dilakukan kepada pegawai Dinas Kearsipan dan Perpustakaan Kota Mataram untuk memperkenalkan penggunaan sistem *file sharing* berbasis *Nextcloud* serta mekanisme otomatisasi yang telah diterapkan. Kegiatan meliputi penjelasan fitur utama *Nextcloud*, cara mengakses dan berbagi *file*, serta pemanfaatan sistem otomatisasi untuk mendukung pengelolaan arsip digital.
  - Evaluasi: Tahap evaluasi dilakukan untuk menilai efektivitas penerapan *Nextcloud* dan sistem otomatisasi setelah digunakan oleh pegawai. Evaluasi dilakukan melalui observasi penggunaan sistem dan wawancara singkat terkait kemudahan penggunaan, efisiensi kerja, serta manfaat sistem terhadap pengelolaan arsip digital di lingkungan instansi.

## 4. HASIL DAN PEMBAHASAN

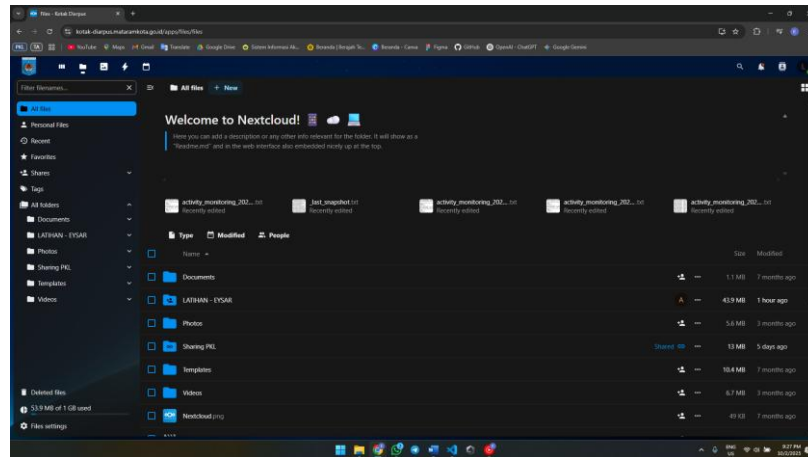
### 4.1. Analisis *Nextcloud*

*Nextcloud* yang digunakan dalam kegiatan ini merupakan sistem *file sharing* berbasis *open source* yang telah diterapkan di lingkungan Dinas Kearsipan dan Perpustakaan Kota Mataram dengan nama layanan Kotak Diarpus. Karena sistem telah tersedia sebelumnya, tahap analisis difokuskan pada pemahaman struktur penyimpanan dan mekanisme berbagi *file* antarbidang. Pengguna dapat mengakses sistem melalui halaman *login* resmi menggunakan akun internal, sehingga hanya pengguna terdaftar yang memiliki akses terhadap arsip digital instansi.



Gambar 4.1 Halaman *Dashboard* Kotak Diarpus

Setelah berhasil melakukan *login*, pengguna akan diarahkan ke *dashboard* utama *Nextcloud* yang berfungsi sebagai pusat navigasi pengelolaan arsip digital. Pada sisi kiri terdapat panel “*Recommended Files*” yang menampilkan dokumen yang sering diakses atau direkomendasikan untuk memudahkan pencarian *file* penting. Sementara itu, sisi kanan menampilkan panel “*Upcoming Events*” berisi jadwal kegiatan mendatang atau keterangan kosong jika tidak ada agenda. Di bagian atas, tersedia menu navigasi untuk mengakses berbagai fitur seperti penyimpanan *file*, galeri, aktivitas, dan pengaturan akun.

Gambar 4.2 Halaman *Files* Kotak Diarpus

Halaman ini menampilkan tampilan utama penyimpanan *file Nextcloud* setelah pengguna berhasil *login*. Pada sisi kiri terdapat panel navigasi berisi menu seperti “All files”, “Personal Files”, “Recent”, “Favorites”, dan “Shares” untuk memudahkan akses ke kategori *file* tertentu. Bagian tengah menampilkan daftar *folder* dan dokumen seperti “Documents”, “LATIHAN – EYSAR”, “Photos”, “Sharing”, “Templates”, dan “Videos”, lengkap dengan informasi ukuran, waktu modifikasi terakhir, serta status berbagi. Tata letak *folder* yang terstruktur ini memudahkan pengguna dalam mengelola dan mengakses arsip digital secara efisien.

#### 4.2. Pembuatan Skrip Otomatisasi

Untuk meningkatkan efisiensi pengelolaan arsip digital, dikembangkan sistem otomatisasi berbasis *Python* yang diintegrasikan dengan layanan *Task Scheduler* pada sistem operasi Windows. Proses otomatisasi ini memungkinkan aktivitas seperti pencadangan, pengarsipan, dan pemantauan *file* dilakukan secara terjadwal tanpa keterlibatan langsung pengguna.

Fokus utama dari sistem otomatisasi ini adalah pembuatan skrip *auto backup*, yang bertugas membuat salinan arsip digital dari direktori *Nextcloud* secara berkala. Proses ini dilakukan dengan mengompresi seluruh isi *folder* sumber ke dalam satu file cadangan (.zip), serta menghapus *file backup* lama secara otomatis untuk menghemat ruang penyimpanan. Implementasi ini menggunakan pustaka standar *Python* seperti “os”, “zipfile”, dan “datetime”.

```

17 def create_backup():
18     backup_file = generate_backup_file()
19     print(f"[INFO] Membuat backup dari '{NEXTCLOUD_FOLDER}' ke '{backup_file}'")
20
21     with zipfile.ZipFile(backup_file, 'w', zipfile.ZIP_DEFLATED) as zipf:
22         for root, dirs, files in os.walk(NEXTCLOUD_FOLDER):
23             if os.path.abspath(root).startswith(os.path.abspath(BACKUP_FOLDER)):
24                 continue
25
26             for file in files:
27                 full_path = os.path.join(root, file)
28                 relative_path = os.path.relpath(full_path, NEXTCLOUD_FOLDER)
29                 zipf.write(full_path, relative_path)
30
31     print(f"[INFO] Backup selesai: {backup_file}")

```

Gambar 4.3 Potongan Skrip Otomatisasi

Potongan skrip pada gambar menunjukkan fungsi “create\_backup()” yang berfungsi untuk membuat salinan cadangan (*backup*) dari seluruh data yang tersimpan di direktori *Nextcloud*. Fungsi ini diawali dengan pembuatan nama *file backup* menggunakan “generate\_backup\_file()”, kemudian mencetak informasi proses pencadangan yang sedang berlangsung. Proses *backup* dilakukan dengan cara mengompres seluruh *file* dan *folder* dari direktori “NEXTCLOUD\_FOLDER” ke dalam satu *file* arsip berekstensi “.zip” menggunakan modul “zipfile”. Agar tidak terjadi pencadangan berulang di dalam *folder backup* itu sendiri, skrip memeriksa jalur direktori dengan memastikan bahwa *file* yang sedang dikompres bukan bagian dari “BACKUP\_FOLDER”. Setiap *file* dalam direktori *Nextcloud* akan dimasukkan ke dalam arsip dengan struktur relatif yang sama seperti aslinya. Setelah proses selesai, sistem menampilkan pesan bahwa *backup* telah berhasil dibuat beserta lokasi *file* hasilnya.

Selain skrip *auto backup*, terdapat skrip lainnya seperti *auto zip & extract* untuk kompresi file berukuran besar serta *activity monitoring* untuk mencatat perubahan file (penambahan, penghapusan, dan modifikasi). Semua skrip memanfaatkan pustaka *Python* seperti “os”, “zipfile”, dan “datetime”, serta ditulis secara modular agar mudah dikembangkan kembali.

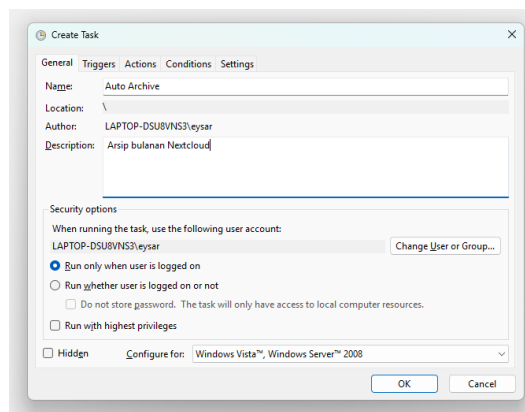
### 4.3. Penerapan Integrasi Task Scheduler

*Task Scheduler* dimanfaatkan untuk mengotomatisasi eksekusi berbagai skrip otomatisasi yang telah dikembangkan, seperti skrip *auto zip & extract*, *auto backup*, serta monitoring aktivitas, sehingga seluruh proses dapat berjalan secara terjadwal tanpa keterlibatan langsung pengguna. Proses penjadwalan dilakukan dengan memanfaatkan *file batch* (.bat) sebagai perantara untuk menjalankan skrip *Python*, karena *Task Scheduler* tidak dapat mengeksekusi *file* “.py” secara langsung.

```
@echo off
echo [MULAI] Menjalankan proses auto backup pada Nextcloud...
echo.
python "C:\Users\eyсар\Documents\Kuliah\Praktik Kerja Lapangan\Nextcloud\Scripts\auto_backup.py"
echo.
echo [SELESAI] Proses selesai.
pause
```

Gambar 4.4 Contoh Isi Skrip *Batch*

Skrip *batch* berfungsi sebagai perantara untuk menjalankan berbagai skrip otomatisasi *Nextcloud* secara terjadwal melalui *Task Scheduler* tanpa memerlukan interaksi manual. Skrip ini mengeksekusi *file Python* terkait, menampilkan status proses di *Command Prompt*, dan memberi notifikasi setelah selesai. Selain fleksibel untuk digunakan pada berbagai proses seperti *auto archive*, *auto backup*, dan monitoring aktivitas, skrip *batch* juga dikonfigurasi di *Task Scheduler* melalui pembuatan *task* baru, penentuan jadwal eksekusi (*trigger*), serta aksi (*action*) untuk menjalankannya secara otomatis sesuai waktu yang ditentukan.



Gambar 4.5 Jendela *Create Task*

Tahap awal dalam konfigurasi *Task Scheduler* dilakukan dengan membuat *task* baru melalui jendela *Create Task*, seperti ditunjukkan pada gambar. Pada tahap ini, pengguna dapat memberikan nama tugas, menentukan akun eksekusi, serta mengatur deskripsi dan izin akses sesuai kebutuhan. Selanjutnya, dilakukan pengaturan jadwal (*trigger*) untuk menentukan waktu dan frekuensi eksekusi, misalnya harian, mingguan, atau bulanan, disesuaikan dengan kebutuhan operasional sistem. Setelah jadwal ditentukan, pengguna menambahkan *action* berupa perintah untuk menjalankan *file batch* (.bat) yang berisi skrip *Python*. Pengaturan tambahan pada bagian “Settings” juga dapat dilakukan untuk memastikan keandalan proses, seperti menjalankan tugas saat jadwal terlewat, mengizinkan eksekusi manual, atau mengatur ulang proses jika terjadi kegagalan.

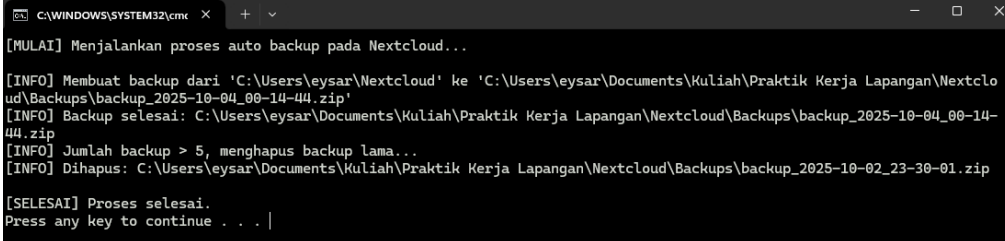
Name	Status	Triggers	Next Run Time	Last Run Time	Last Run Result	Author	Created
Activity Monitoring	Ready	Multiple triggers defined	10/4/2025 12:00:00 PM	10/4/2025 12:07:11 AM	The operation com...	LAPTOP-...	9/27/2025 10:15:27 PM
Auto Archive	Ready	At 11:59 PM on day Last of Janu...	10/31/2025 11:59:00 PM	10/4/2025 12:07:17 AM	The operation com...	LAPTOP-...	10/3/2025 2:40:24 PM
Auto Backup	Ready	At 11:30 PM every day	10/4/2025 11:30:00 PM	10/4/2025 12:07:21 AM	The operation com...	LAPTOP-...	6/21/2025 10:21:32 PM
Auto Unzip	Ready	At 1:00 AM on day 1 of January, ...	11/1/2025 1:00:00 AM	10/4/2025 12:07:26 AM	The operation com...	LAPTOP-...	9/28/2025 7:29:33 PM

Gambar 4.6 Hasil *Create Task*

Gambar di atas menampilkan daftar tugas otomatis (*scheduled tasks*) yang telah berhasil dibuat di dalam *Task Scheduler*. Terlihat empat tugas utama yaitu “Activity Monitoring”, “Auto Archive”, “Auto Backup”, dan “Auto Unzip”, masing-masing dengan status “Ready” yang menandakan siap untuk dijalankan sesuai jadwal yang telah ditentukan. Kolom “Triggers” menunjukkan waktu atau kondisi pemicu setiap tugas, misalnya tugas “Auto Backup” dijadwalkan berjalan setiap hari pukul 23:30. Sementara itu, kolom “Next Run Time” menunjukkan waktu eksekusi berikutnya, dan “Created” mencatat tanggal pembuatan masing-masing tugas.

#### 4.4. Hasil Pengujian Otomatisasi

Pengujian dilakukan untuk memastikan bahwa konfigurasi otomatisasi yang telah dibuat melalui *Task Scheduler* dapat berjalan sesuai dengan tujuan masing-masing. Proses pengujian dilakukan secara manual dengan menjalankan setiap *task* secara langsung untuk memverifikasi apakah skrip *batch* dapat dieksekusi dan menghasilkan keluaran sesuai yang diharapkan.



```

C:\WINDOWS\SYSTEM32\cmd X
[MULAI] Menjalankan proses auto backup pada Nextcloud...
[INFO] Membuat backup dari 'C:\Users\eyasar\Nextcloud' ke 'C:\Users\eyasar\Documents\Kuliah\Praktik Kerja Lapangan\Nextcloud\Backups\backup_2025-10-04_00-14-44.zip'
[INFO] Backup selesai: C:\Users\eyasar\Documents\Kuliah\Praktik Kerja Lapangan\Nextcloud\Backups\backup_2025-10-04_00-14-44.zip
[INFO] Jumlah backup > 5, menghapus backup lama...
[INFO] Dihapus: C:\Users\eyasar\Documents\Kuliah\Praktik Kerja Lapangan\Nextcloud\Backups\backup_2025-10-02_23-30-01.zip
[SELESAI] Proses selesai.
Press any key to continue . . . |
  
```

Gambar 4.7 Hasil Pengujian *Task Auto Backup*

Gambar tersebut menunjukkan hasil pengujian dari task “Auto Backup” yang dijalankan secara manual melalui *Task Scheduler*. Berdasarkan *output* yang ditampilkan, proses pencadangan berhasil dilakukan dengan membuat *file backup* dari direktori sumber ke lokasi tujuan yang telah ditentukan, disertai dengan penamaan *file* yang mencantumkan informasi waktu pembuatan. Selain itu, sistem juga melakukan manajemen *backup* dengan menghapus cadangan lama ketika jumlah *backup* melebihi batas yang ditetapkan. Seluruh proses berjalan dengan baik tanpa *error* dan diakhiri dengan notifikasi bahwa proses telah selesai, menandakan bahwa skrip *batch* dan konfigurasi *Task Scheduler* untuk fitur *auto backup* berfungsi sebagaimana mestinya.

Hasil pengujian menunjukkan bahwa semua proses berjalan dengan baik tanpa *error*. Skrip *auto zip & extract* berhasil mengarsipkan dan mengekstrak file sesuai jadwal, *auto backup* menciptakan *file* cadangan lengkap dengan nama *file* berbasis waktu, sementara *activity monitoring* mampu mencatat seluruh perubahan *file* secara detail, meliputi waktu kejadian, nama *file*, dan jenis aktivitas.

Secara keseluruhan, penerapan *Nextcloud* yang dikombinasikan dengan sistem otomatisasi berbasis *Python* dan *Task Scheduler* terbukti mampu meningkatkan efisiensi, keamanan, serta keandalan pengelolaan arsip digital di lingkungan Dinas Kearsipan dan Perpustakaan Kota Mataram. Sistem ini berhasil mengurangi beban kerja manual, mempercepat proses pertukaran data, serta menjaga konsistensi dan keutuhan arsip digital secara berkelanjutan.

#### 4.5. Evaluasi Pengguna dan Dampak terhadap Mitra

Setelah implementasi sistem dilakukan, kegiatan dilanjutkan dengan sosialisasi dan pendampingan penggunaan *Nextcloud* kepada pegawai Dinas Kearsipan dan Perpustakaan Kota Mataram. Pegawai diberikan penjelasan mengenai cara mengakses sistem Kotak Diarpus, melakukan pengelolaan *file*, berbagi dokumen antarbidang, serta memahami mekanisme otomatisasi yang berjalan pada sistem.



Gambar 4.8 Kegiatan Pendampingan dan Evaluasi Penggunaan Sistem

Berdasarkan hasil observasi dan wawancara singkat terhadap pengguna, penerapan *Nextcloud* memberikan dampak positif terhadap proses pengelolaan arsip digital di lingkungan instansi. Pengguna menyatakan bahwa proses pertukaran *file* menjadi lebih mudah, cepat, dan terstruktur dibandingkan sebelumnya. Selain itu, penggunaan sistem *file sharing* terpusat membantu pegawai mengakses dokumen dari berbagai perangkat tanpa harus menggunakan media penyimpanan fisik secara langsung.

Penerapan sistem otomatisasi berbasis *Python* dan *Task Scheduler* juga memberikan manfaat dalam mengurangi pekerjaan rutin yang sebelumnya dilakukan secara manual, seperti proses pencadangan data dan pengarsipan *file*. Dengan adanya *backup* otomatis dan monitoring aktivitas *file*, pegawai merasa lebih terbantu dalam menjaga keamanan dan keberlanjutan arsip digital.

Secara keseluruhan, kegiatan pengabdian ini memberikan manfaat bagi mitra dalam meningkatkan efisiensi kerja, keteraturan pengelolaan arsip, serta keamanan data digital di lingkungan Dinas Kearsipan dan Perpustakaan Kota Mataram.

## 5. KESIMPULAN DAN SARAN

### 5.1. Kesimpulan

Berdasarkan hasil kegiatan pengabdian masyarakat yang telah dilakukan, penerapan *Nextcloud* sebagai sistem *file sharing* terpusat bernama Kotak Diarpus berhasil membantu proses pertukaran dan pengelolaan arsip digital di Dinas Kearsipan dan Perpustakaan Kota Mataram menjadi lebih terstruktur, aman, dan mudah diakses. Integrasi sistem otomatisasi berbasis *Python* dan *Task Scheduler* juga mampu menjalankan proses rutin seperti pencadangan, kompresi *file*, ekstraksi *file*, dan monitoring aktivitas secara otomatis tanpa intervensi langsung pengguna.

Hasil implementasi dan pengujian menunjukkan bahwa sistem berjalan dengan baik dan mampu meningkatkan efisiensi operasional dalam pengelolaan arsip digital. Selain itu, kegiatan sosialisasi dan pendampingan yang dilakukan kepada pegawai membantu meningkatkan pemahaman pengguna terhadap pemanfaatan sistem *cloud storage* berbasis *open source* dalam mendukung administrasi digital instansi.

### 5.2. Saran

Agar hasil analisis ini dapat dijadikan dasar pengembangan lebih lanjut, beberapa saran yang dapat dipertimbangkan antara lain melakukan penyesuaian antarmuka *Nextcloud* agar lebih selaras dengan identitas instansi dan kebutuhan pengguna, memindahkan skrip otomatisasi ke sisi server untuk meningkatkan keandalan sistem tanpa bergantung pada perangkat pengguna, serta mengembangkan sistem notifikasi otomatis berbasis *email* atau *desktop alert* guna memberikan informasi secara *real-time* terkait hasil proses kompresi, pencadangan, dan aktivitas *file*.

## UCAPAN TERIMA KASIH

Puji dan syukur penulis panjatkan ke hadirat Allah SWT, karena berkat rahmat dan karunia-Nya, penulis dapat menyelesaikan kegiatan pengabdian masyarakat ini dengan baik dan tersusun hingga tuntas. Penulis juga menyampaikan terima kasih kepada Dinas Kearsipan dan Perpustakaan Kota Mataram atas kesempatan belajar secara langsung di lingkungan kerja. Tidak lupa, penulis juga mengucapkan terima kasih kepada keluarga dan teman-teman yang selama ini telah mendukung dan memberikan semangat serta doa.

## DAFTAR PUSTAKA

- [1] D. Darmawan, "NEXTCLOUD: KEAMANAN DATA TERBAIK DENGAN MANAJEMEN FILE DAN PENGGUNA YANG CERDAS," *Jurnal Sosial dan Teknologi (SOSTECH)*, vol. 4, no. 1, pp. 80–89, Jan. 2024.
- [2] T. Dealva Arsyad, I. Agi Berutu, K. Rifail Azis, N. Talitha Jehian, and D. Kiswanto, "IMPLEMENTASI FILE SERVER BERBASIS CLOUD MENGGUNAKAN LINUX UBUNTU 24.04 DAN NEXTCLOUD," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 8, no. 6, pp. 12595–12600, Dec. 2024.
- [3] Gufron and P. Sri Syahriarti, "Perancangan Private Cloud Storage dan Cloud Office Menggunakan OnlyOffice dan Nextcloud Pada Pusat Pelatihan Djamboe Training Center," *Jiic: Jurnal Intelek Insan Cendikia*, vol. 1, no. 6, pp. 2120–2128, Aug. 2024, [Online]. Available: <https://jicnusantara.com/index.php/jiic>
- [4] M. H. Maulana, "Python Bahasa Pemrograman Yang Ramah Bagi Pemula," *JISCO (Journal of Information System and Computing)*, vol. 2, no. 2, pp. 73–78, Dec. 2024.
- [5] I. F. Al-Hadi, S. Ilham, C. Chusna, and Abd. C. Fauzan, "Implementasi Penjadwalan Round Robin pada Task Scheduler untuk Pembaruan Aplikasi Otomatis," *ILKOMNIKA: Journal of Computer Science and Applied Informatics*, vol. 1, no. 1, pp. 11–14, Aug. 2019, doi: 10.28926/ilkomnika.v1i1.13.
- [6] F. Nahla, S. R. Zulaikha, and Asnawi, "PEMANFAATAN CLOUD COMPUTING UNTUK MENINGKATKAN EFISIENSI DAN SKALABILITAS SISTEM INFORMASI PERPUSTAKAAN DIGITAL," *Indonesian Journal of Library and Information Science*, vol. 6, no. 1, pp. 51–57, Jun. 2025.
- [7] M. S. Rumetna, "PEMANFAATAN CLOUD COMPUTING PADA DUNIA BISNIS: STUDI LITERATUR," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 5, no. 3, pp. 305–314, Aug. 2018, doi: 10.25126/jtiik.201853595.

- [8] R. Wiji Wahyuningrum, E. Krisnaningsih, A. Dedi Jubaedi, F. Akbar, and S. Dwiyatno, "PEMANFAATAN CLOUD COMPUTING UNTUK PENGELOLAAN ADMINSTRASI SEKOLAH MENGGUNAKAN METODE NEXTCLOUD," *Jurnal ProTekInfo*, vol. 11, no. 2, pp. 28–35, Aug. 2024.
- [9] A. Albert, J. P. Gunawan, D. S. Wijaya, and J. Jonathan, "Penggunaan Nextcloud untuk Distribusi Data dalam Lingkup Perusahaan," vol. 12, no. 2, pp. 1–8, Jul. 2024.
- [10] A. T. Atmaja, D. Santoso, and P. Ninghardjanti, "PENERAPAN SISTEM OTOMATISASI ADMNISTRASI UNTUK MENINGKATKAN EFEKTIVITAS DAN EFISIENSI KERJA DI BIDANG PENDAPATAN DINAS PERDAGANGAN KOTA SURAKARTA," *Jurnal Informasi dan Komunikasi Administrasi Perkantoran*, vol. 2, no. 2, pp. 1–14, Feb. 2018.
- [11] M. F. Jambak, S. Kadarwati, and H. Supratikta, "FUNGSI OTOMATISASI PERKANTORAN DAN SISTEM INFORMASI MANAJEMEN TERHADAP PENINGKATAN PRODUKTIVITAS KARYAWAN BADAN WAKAF AL-QUR'AN," *Multidisciplinary Indonesian Center Journal (MICJO)*, vol. 1, no. 3, pp. 1237–1247, Jul. 2024, doi: 10.62567/micjo.v1i3.147.
- [12] J. N. Semendawai, I. Febiola, B. Pamungkas, and M. D. Ruliansyah, "Perancangan Aplikasi Otomatisasi Menggunakan Bahasa Pemrograman Python Pada Aktivitas Monitoring Pemakaian Data Harian Kartu Internet Of Things," *Jurnal Rekayasa Elektro Sriwijaya*, vol. 3, no. 1, pp. 193–4063198, Nov. 2021.
- [13] N. Setyawati, G. D. K. Ningrum, and F. Alqodri, "PENERAPAN TASK SCHEDULER UNTUK OPTIMALISASI LAPORAN PERTANGGUNGJAWABAN KEUANGAN DI KEMAHASISWAAN FAKULTAS TEKNIK UNIVERSITAS NEGERI MALANG," *SEMINASTIKA*, vol. 5, no. 1, Aug. 2024.