

BAB IV

HASIL DAN PEMBAHASAN

4.1 Profil Organisasi

4.1.1 Gambaran Ringkas Organisasi

Lembaga kearsipan di Kabupaten Cilacap memiliki sejarah yang panjang dan dinamis. Keberadaannya berawal dari Subbag Tata Usaha pada bagian umum Sekretariat Daerah Cilacap. Seiring berjalannya waktu, lembaga ini mengalami beberapa perubahan penting. Pada tahun 1998, terjadi perubahan signifikan ketika Kantor Arsip Daerah Kabupaten Cilacap didirikan. Ini merupakan langkah maju yang penting dalam pengelolaan arsip di daerah tersebut. Perubahan terus terjadi. Pada tahun 2000, lembaga ini berubah menjadi Kantor Arsip dan Perpustakaan Daerah Kabupaten Cilacap sehingga menunjukkan peningkatan fokus pada pendidikan dan penyebaran informasi. Pada tahun 2003, lembaga ini kembali berubah menjadi Sub Bagian Arsip pada bagian Umum Sekretariat Daerah Cilacap. Perubahan ini menunjukkan penyesuaian struktur organisasi untuk memenuhi kebutuhan yang berkembang.

Pada tahun 2007, lembaga ini kembali menjadi Kantor Arsip dan Perpustakaan Daerah Kabupaten Cilacap, menunjukkan kembali penekanan pada peran penting perpustakaan dalam masyarakat. Pada tahun 2010, Kantor Arsip dan Perpustakaan Daerah Kabupaten Cilacap bergabung dengan Badan Pendidikan Pelatihan Arsip dan Perpustakaan Daerah Kabupaten Cilacap. Ini menunjukkan upaya untuk memperkuat pendidikan dan pelatihan dalam bidang arsip dan perpustakaan. Akhirnya, pada tahun 2017, Badan Pendidikan Pelatihan Arsip dan Perpustakaan berubah menjadi Dinas Kearsipan dan Perpustakaan Kabupaten Cilacap. Perubahan ini mencerminkan peningkatan status dan peran lembaga ini dalam masyarakat.

4.1.2 Tugas dan Fungsi

a. Tugas

Berdasarkan Peraturan Bupati Cilacap Nomor 156 Tahun 2021 pasal 4, Dinas Kearsipan dan Perpustakaan mempunyai tugas membantu Bupati dalam melaksanakan urusan pemerintahan bidang kearsipan dan perpustakaan yang menjadi kewenangan daerah dan tugas pembantuan yang ditugaskan kepada daerah.

b. Fungsi

Berdasarkan Peraturan Bupati Cilacap Nomor 156 tahun 2021 pasal 5, Dinas Kearsipan dan Perpustakaan menyelenggarakan fungsi :

- 1) Perumusan kebijakan di bidang kearsipan dan perpustakaan.
- 2) Pelaksanaan koordinasi kebijakan di bidang kearsipan dan perpustakaan.
- 3) Pelaksanaan kebijakan di bidang kearsipan dan perpustakaan.
- 4) Pelaksanaan evaluasi dan pelaporan di bidang kearsipan dan perpustakaan.
- 5) Pelaksanaan fungsi kesekretariatan Dinas.
- 6) Pelaksanaan fungsi lain yang diberikan oleh pimpinan terkait dengan tugas dan fungsinya.

4.1.3 Visi dan Misi

a. Visi

Cilacap Yang Berbudaya dan Sejahtera.

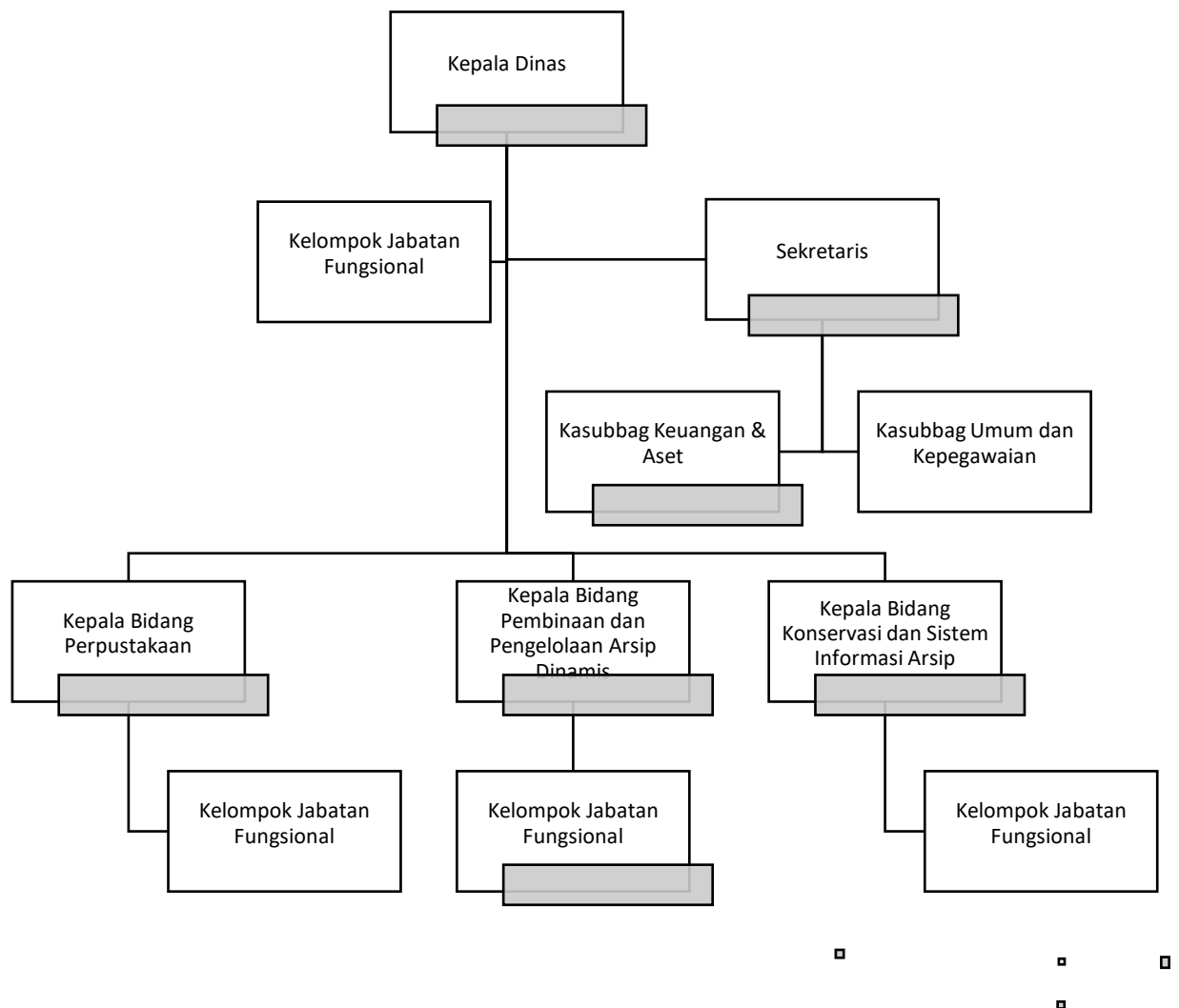
b. Misi

- 1) Mewujudkan Masyarakat yang menjunjung tinggi budaya dan nilai-nilai luhur serta kerukunan antar umat beragama

ditandai dengan hubungan antar Masyarakat yang harmonis dan saling menghormati.

- 2) Menciptakan pemerintahan yang baik dan terpercaya dalam rangka memberikan pelayanan kepada Masyarakat.
- 3) Penegakan supremasi hukum dalam rangka menjunjung tinggi kebenaran dan keadilan.
- 4) Menciptakan stabilitas politik dan keamanan sebagai pendukung iklim investasi dan peningkatan perekonomian daerah.
- 5) Memaksimalkan potensi daerah melalui peningkatan daya saing yang andal.
- 6) Meningkatkan kualitas ekonomi rakyat melalui peningkatan kualitas *Sumber Daya Manusia* (SDM) dan pemanfaatan *Sumber Daya Alam* (SDA) yang ada dengan memperhatikan aspek keseimbangan pelestarian alam.
- 7) Membangun infrastruktur yang dapat menunjang Pembangunan.
- 8) Mewujudkan Pembangunan yang memperhatikan aspek tata ruang yang menunjang kesejahteraan sesuai dengan Peraturan Daerah Nomor 6 Tahun 2004 tentang Rencana Tata Ruang Wilayah Kabupaten Cilacap yang telah ditetapkan.
- 9) Meningkatkan partisipasi Masyarakat dalam proses Pembangunan.

4.1.4 Struktur Organisasi



Bidang Kearsipan

Bidang Kearsipan mempunyai tugas perumusan dan pelaksanaan kebijakan, pengoordinasian, pemantauan, evaluasi serta pelaporan meliputi pengelolaan arsip dinamis aktif dan arsip statis, pembinaan kearsipan, pengawasan kearsipan, serta pelayanan dan pemanfaatan kearsipan.

4.2 Pembuatan Sistem

4.2.1 Membuat format *SpreadSheet* atau format excel

Tahapan awal dalam membuat sstem otomasi yang digunakan untuk kegiatan pemusnahan arsip terutama pada kegiatan pencarian arsip yang habis masa retensinya adalah membuat format *SpreadSheet*. Dalam proses ini, penting untuk merancang struktur *SpreadSheet* dengan cermat agar semua informasi yang diperlukan dapat tercatat dengan baik.

SpreadSheet digunakan sebagai pengganti Excel, yang mungkin dianggap lebih rumit oleh beberapa pengguna. Keunggulan utama dari *SpreadSheet* adalah kemudahan akses dan kemampuannya untuk berbagi data secara *real-time*. Selain itu, pengelolaan data pada *SpreadSheet* dapat terintegrasi dengan perangkat lain, khususnya platform Telegram, sehingga mempermudah pengguna dalam melakukan pemusnahan arsip secara efektif dan efisien.

Integrasi ini juga memungkinkan pengguna untuk berinteraksi langsung dengan data yang ada, meningkatkan produktivitas dan akurasi dalam kegiatan pemusnahan.

DAFTAR ARSIP INAKTIF									
SEKRETARIAT DINAS KEARSIPAN DAN PEPRUSTAKAAN									
KABUPATEN CILACAP									
TAHUN PENATAAN 2024									
NO.	URAIAN MASALAH	BULAN & TAHUN PENCIPTAAN	JUMLAH BERKAS	NO. SEMENTARA	KLASIFIKASI ARSIP	NILAI RETENSI	DAFTAR PESTEROGAN		TAMBAHAN
							ARSIP	RETIKSI	
1	Rencana Kerja Pemerintah Daerah (RKPD) Tahun 2016	Juni 2016	1	0.001	Buku	2	7		
2	Daftar Rincian Kerja Dinas Kearsipan dan Peprustakaan Kabupaten Cilacap Tahun 2016	Juni 2016	1	0.002	Buku	2	7		
3	Rencana Kerja Dinas Kearsipan dan Peprustakaan Kabupaten Cilacap Tahun 2017	Juni 2017	1	0.003	Buku	2	7		
4	Survei Arsip Daerah (SAD) Kabupaten Cilacap Tahun 2017	Desember 2017	1	0.004	Buku	2	7		
5	Survei Arsip Daerah (SAD) Kabupaten Cilacap Tahun 2018	Desember 2018	1	0.005	Buku	2	7		
6	Survei Arsip Daerah (SAD) Kabupaten Cilacap Tahun 2019	Desember 2019	1	0.006	Buku	2	7		
7	Survei Arsip Daerah (SAD) Kabupaten Cilacap Tahun 2020	Desember 2020	1	0.007	Buku	2	7		
8	Survei Arsip Daerah (SAD) Kabupaten Cilacap Tahun 2021	Desember 2021	1	0.008	Buku	2	7		
9	Survei Arsip Daerah (SAD) Kabupaten Cilacap Tahun 2022	Desember 2022	1	0.009	Buku	2	7		
10	Survei Arsip Daerah (SAD) Kabupaten Cilacap Tahun 2023	Desember 2023	1	0.010	Buku	2	7		

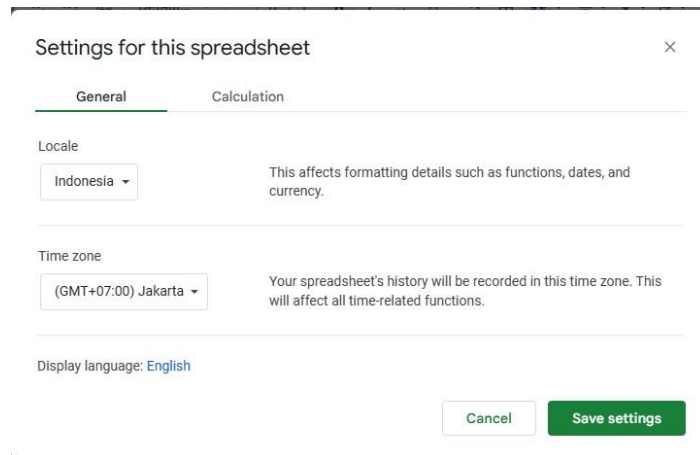
Gambar 4. 1 Format Tabel Ms Excel

Dalam membuat tabel format *SpreadSheet*, terdapat beberapa ketentuan yang harus diperhatikan agar informasi yang disajikan menjadi lebih terstruktur dan mudah dipahami. Ketentuan-ketentuan tersebut antara lain adalah sebagai berikut:

1. Kolom pertama harus mencantumkan Nomor, yang berfungsi sebagai urutan identifikasi setiap entri dalam tabel.
4. Kolom kedua berisikan Nomor Klasifikasi Arsip, yang penting untuk pengelompokan arsip berdasarkan kategori tertentu.
5. Kolom ketiga berisi Uraian Masalah Arsip, di mana informasi detail mengenai isi arsip dapat dijelaskan secara jelas.
6. Kolom keempat berisi Bulan dan Tahun penciptaan arsip. Di bawah kolom ini, terdapat tiga kolom tambahan yang mencakup “Bulan, Tahun, dan Bulan dalam format angka” untuk mempermudah pencarian berdasarkan waktu.
7. Kolom kelima menampung informasi tentang Jumlah berkas, yang menunjukkan banyaknya dokumen dalam setiap kategori.
8. Kolom keenam berisi No. Sementara, yang dapat digunakan untuk mencatat nomor sementara sebelum arsip dilakukan pengklasifikasian secara permanen.

- 9 Kolom ketujuh mencakup Klasifikasi Keamanan, yang menunjukkan tingkat kerahasiaan dari arsip tersebut.
- 10 Kolom kedelapan berisi Hak Akses, sehingga pengguna dapat mengetahui siapa saja yang diperbolehkan mengakses informasi dalam arsip tersebut.
- 11 Kolom kesembilan berisi Dasar Pertimbangan, yang memberikan penjelasan mengenai alasan pengklasifikasian arsip.
- 12 Kolom kesepuluh berisi Masa Retensi, yang mencakup dua kolom lagi yaitu masa retensi aktif dan masa retensi inaktif untuk mengatur periode penyimpanan arsip.
- 13 Kolom kesebelas disiapkan untuk Tahun Musnah Arsip, yang menunjukkan kapan arsip akan dihancurkan.
- 14 Kolom kedua belas menjelaskan tanggal habis retensi arsip, untuk memastikan bahwa arsip tidak disimpan lebih lama dari yang diperlukan.
- 15 Terakhir, kolom ketiga belas berisi Keterangan, yang dapat dimanfaatkan untuk memberikan informasi tambahan yang relevan mengenai arsip.

4.2.2 Mengganti Bahasa *SpreadSheet* ke Bahasa Indonesia



Gambar 4. 2 Ubah Bahasa ke Bahasa Indonesia

Langkah berikutnya adalah melakukan perubahan dari bahasa *SpreadSheet* ke Bahasa Indonesia dan mengubah *time zone* ke waktu Jakarta (GMT+07:00). Perubahan ini bertujuan agar memudahkan dalam penggunaan rumus dan fungsi yang akan digunakan untuk menghitung tanggal dan tahun yang tertera dalam arsip. Proses penggantian bahasa dan *time zone* pada *SpreadSheet* dapat dilakukan dengan mengakses *menu File* dan kemudian masuk ke opsi *setting*. Dengan melakukan perubahan ini, diharapkan pengguna dapat melakukan pengolahan data dengan lebih efisien dan akurat sesuai dengan kebutuhan yang ada.

4.2.3 Memasukkan Rumus *Month*

NO	KLASIFIKASI	URAIAN MASALAH	BULAN & TAHUN PENCIPTAAN			JUMLAH BERKAS	NO SEMENTARA
1	050	Rencana Kerja Perubahan Satuan Kerja Perangkat Daerah (Renja SKPD) Tahun 2016 Bandiklat, Arsip dan Peprusa	Juni	2016		1	R. 047
2	050	Copy Rencana Kerja Dinas Kearsipan dan Perpustakaan Kabupaten Cilacap Tahun 2018	Juni	2017		1	R.012
3	050	Rencana Kerja Dinas Kearsipan dan Perpustakaan Kabupaten Cilacap Tahun 2018	Juni	2017		1	R.045
4	050	Laporan Akuntabilitas Kinerja Instansi Pemerintah (LKJIIP) Periode Tahun 2018 Dinas Kearsipan dan Perpustakaan Kabupaten Cilacap	Desember	2018		1	R.025
5	050	Data Dukung Indikator Kinerja Kunci (IKK) Lampiran III.2 Dinas Kearsipan dan Perpustakaan Kabupaten Cilacap Urusan Kearsipan dan Urusan Perpustakaan - Laporan Penyelenggaraan Pemerintahan Daerah Kabupaten Cilacap Tahun 2018	Juli	2018		2	R.028
6	050	Rencana Kerja dan Rancangan Akhir Rencana Kerja Dinas Kearsipan dan Perpustakaan Kabupaten Cilacap Tahun 2020	Juli	2018		2	R.023

+ ≡ **Daftar_arsip_Inaktif** ▾ Arsip_Habis_Retensi ▾

Gambar 4. 3 Rumus *Month*

Tahapan selanjutnya dalam proses ini adalah memasukkan rumus *Month* atau bulan. Rumus ini memiliki fungsi penting, yaitu untuk mengubah nama bulan yang masih berada dalam format kalimat menjadi representasi angka. Proses ini sangat krusial, sebab banyak rumus dan fungsi lain yang hanya dapat memproses data dalam format angka. Oleh karena itu, kita perlu memastikan bahwa bulan sudah dikonversi menjadi angka sehingga bisa dipakai dalam perhitungan lebih lanjut.

Di *spreadSheet* yang kita gunakan, terdapat kolom kosong di bawah judul Bulan & Tahun Penciptaan. Kolom kosong ini memang diperuntukkan khusus untuk memasukkan rumus *Month*. Dengan menggunakan rumus ini, kita bisa mengubah nama bulan yang tertulis menjadi bentuk angka yang lebih mudah diolah. Berikut adalah rumus *Month* yang bisa kita gunakan: (=MONTH(1& kolom bulan yang ingin diubah menjadi angka)).

Sebagai contoh, mari kita lihat gambar di atas. Di situ, kita ambil contoh kolom bulan yang ingin diubah menjadi angka, yakni bulan Juni. Ketika kita memasukkan rumus sesuai dengan kolom yang ditunjuk, maka akan otomatis muncul angka 6. Hal ini terjadi karena bulan Juni merupakan bulan keenam dalam urutan bulan dalam setahun. Proses ini tidak hanya mempermudah pemahaman tetapi juga mempermudah analisis data lebih lanjut.

4.2.4 Memasukkan Rumus *sum* Pada Kolom Tahun Musnah

BULAN & TAHUN PENCIPTAAN		JUMLAH BERKAS	NO SEMENTARA	KLASIFIKASI KEMANAN	HAK AKSES	DASAR PERTIMBANGAN	MASA RETENSI		TAHUN MUSNAH
							AKTIF	INAKTIF	
Juni	2016	1	R. 047	Biasa	Eselon III	Tidak memiliki dampak yang mengganggu kerja organisasi	2	7	
Juni	2017	1	R.012	Biasa	Eselon III	Tidak memiliki dampak yang mengganggu kerja organisasi	2	7	
Juni	2017	1	R.045	Biasa	Eselon III	Tidak memiliki dampak yang mengganggu kerja organisasi	2	7	
Desember	2018	1	R.025	Biasa	Eselon III	Tidak memiliki dampak yang mengganggu kerja organisasi	2	7	
Juli	2018	2	R.028	Biasa	Eselon III	Tidak memiliki dampak yang mengganggu kerja organisasi	2	7	
Juli	2018	2	R.023	Biasa	Eselon III	Tidak memiliki dampak yang mengganggu kerja organisasi	2	7	

Daftar_arsip_Inaktif ▾
Arsip_Habis_Retensi ▾

Gambar 4. 4 Rumus SUM

Pada tahapan ini, proses untuk mencari Tahun Musnah dapat dilakukan dengan memasukkan rumus *SUM*. Rumus yang digunakan untuk menentukan Tahun Musnah adalah sebagai berikut: $=SUM(\text{kolom tahun penciptaan arsip}; \text{masa retensi aktif arsip}; \text{masa retensi inaktif arsip})$. Dalam konteks ini, kita perlu memahami beberapa komponen yang ada dalam rumus tersebut. Pertama-tama, kita mengambil kolom tahun penciptaan arsip yang berfungsi sebagai titik awal perhitungan.

Sebagai contoh, arsip yang kita cari tahun musnahnya adalah arsip yang diciptakan pada tahun 2016. Selanjutnya, kita harus memperhitungkan masa retensi aktif yang berlaku untuk arsip

tersebut, yang dalam hal ini adalah 2 tahun. Setelah masa retensi aktif, terdapat pula masa retensi inaktif yang perlu diperhatikan, yang dalam contoh ini adalah 7 tahun.

Dengan memasukkan semua informasi tersebut ke dalam rumus, kita dapat menghitung tahun musnah arsip tersebut. Jadi, kita menjumlahkan tahun penciptaan (2016) dengan masa retensi aktif (2 tahun) dan masa retensi inaktif (7 tahun). Dari perhitungan tersebut, kita akan mendapatkan tahun musnah, yang dalam kasus ini adalah 2025.

4.2.5 Mencari Tanggal Habis Retensi dengan Rumus *Date*

BULAN & TAHUN PENCIPTAAN			JUMLAH DOKAS	NO SEMENTARA	KLASIFIKASI KEAMANAN	HAK AKSES	DASAR PERTIMBANGAN	MASA RETENSI		TAHUN MUSNAH	TANGGAL HABIS RETENSI	KETERANGAN
								AKTIF	INAKTIF			
Juni	2016	6	1	R.047	Biasa	Eselon II	Tidak memiliki dampak yang mengganggu kerja organisasi	2	7	2025		Boks 1
Juni	2017	6	1	R.012	Biasa	Eselon II	Tidak memiliki dampak yang mengganggu kerja organisasi	2	7	2025		Boks 1
Juni	2017	6	1	R.045	Biasa	Eselon II	Tidak memiliki dampak yang mengganggu kerja organisasi	2	7	2025		Boks 1
Desember	2018	12	1	R.025	Biasa	Eselon II	Tidak memiliki dampak yang mengganggu kerja organisasi	2	7	2027		Boks 1
Juli	2018	7	2	R.028	Biasa	Eselon II	Tidak memiliki dampak yang mengganggu kerja organisasi	2	7	2027		Boks 1
Juli	2018	7	2	R.023	Biasa	Eselon II	Tidak memiliki dampak yang mengganggu kerja organisasi	2	7	2027		Boks 1
Agustus	2019	8	2	R.002	Biasa	Eselon II	Tidak memiliki dampak yang mengganggu kerja organisasi	2	7	2028		Boks 1

+ ≡ Daftar_arsip_Inaktif ▾ Arsip_Habis_Retensi ▾

Gambar 4. 5 Rumus *Date*

Tanggal habis retensi Arsip dapat dicari menggunakan rumus *Date* dengan menggabungkan bulan habis retensi dan juga tahun habis retensi. Berikut ini adalah rumus *date* untuk mencari Tanggal Habis Retensi ($=DATE(\text{tahun musnah arsip}; \text{bulan penciptaan arsip dalam bentuk angka}; 1)$). Pada gambar di atas, tanggal habis retensi arsip yang dicari adalah arsip yang diciptakan bulan Juni dengan tahun musnah tahun 2025, sehingga tanggal habis retensi arsip tersebut adalah 1 Juni 2025. Angka satu pada rumus *date* digunakan sebagai acuan dikarenakan tidak terdapat keterangan yang pasti kapan

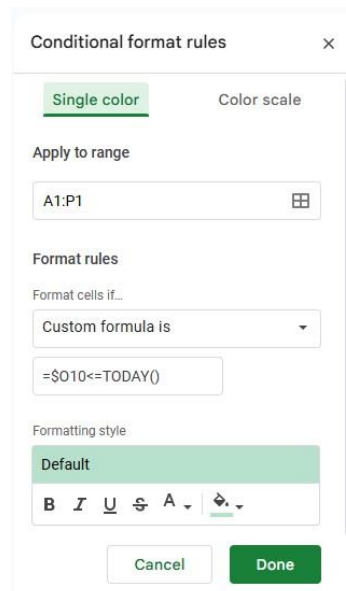
tanggal arsip diciptakan sehingga digunakan tanggal satu. Proses ini penting untuk menjaga arsip-arsip agar tetap teratur dan mudah diakses ketika diperlukan. Dengan cara ini, manajemen arsip dapat berjalan dengan lebih efisien dan efektif karena tanggal habis retensi arsip sudah tercatat dengan jelas. Selain itu, pemakaian rumus *Date* juga memudahkan dalam penjadwalan pemusnahan arsip yang sudah tidak diperlukan agar ruang penyimpanan bisa dimaksimalkan dengan baik. Jadi, penggunaan rumus *Date* untuk mencari Tanggal Habis Retensi Arsip memiliki manfaat yang besar dalam pengelolaan arsip secara keseluruhan.

4.2.6 Mengatur *Conditional formatting*

Conditional formatting merupakan alat yang sangat berguna dalam pengelolaan arsip, terutama dalam menandai arsip-arsip yang sudah habis masa retensinya. Dengan menggunakan fitur ini, pengguna dapat dengan mudah mengidentifikasi dokumen-dokumen yang perlu dihapus atau diarsipkan kembali. Proses ini dilakukan dengan memanfaatkan pemrograman sederhana yang memungkinkan perubahan warna pada arsip secara otomatis. Misalnya, arsip-arsip yang masa retensinya telah lewat Tanggal Habis Retensi akan berubah menjadi warna merah, sehingga menarik perhatian dan memudahkan proses evaluasi.

Selain itu, fitur ini juga dapat disesuaikan lebih lanjut untuk menunjukkan berbagai kondisi lainnya, seperti mendekati tanggal kedaluwarsa atau arsip yang perlu ditindaklanjuti. Dengan demikian, pengguna dapat merencanakan tindakan yang diperlukan dengan lebih efisien. Keberadaan fitur *Conditional formatting* tidak hanya menghemat waktu tetapi juga mengurangi risiko kesalahan manusia dalam proses pengelolaan arsip. Oleh karena itu, penerapan sistem ini sangat dianjurkan bagi perusahaan atau lembaga yang mengelola

volume arsip yang besar, sehingga pengelolaan dokumen dapat berjalan dengan lebih teratur dan efektif.

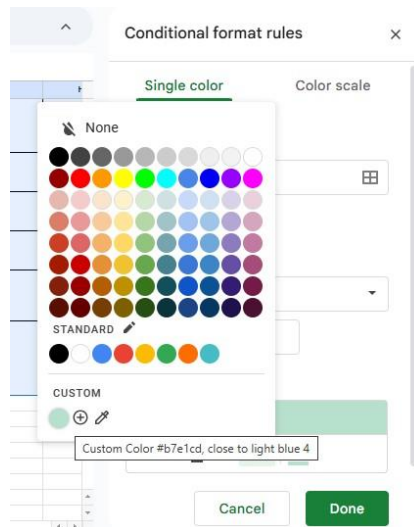


Gambar 4. 6 Conditional Formatting

Berikut adalah langkah-langkah untuk mengatur *Conditional formatting* secara lengkap dan terperinci.

- 1) Pertama, mulailah dengan memblok seluruh tabel yang berisi daftar arsip inaktif. Proses ini dilakukan mulai dari arsip yang bernomor satu hingga bagian pojok kanan bawah dari tabel tersebut. Pastikan bahwa semua sel yang relevan telah dipilih dengan benar untuk memastikan bahwa pengaturan yang akan diterapkan bersifat komprehensif.
- 2) Setelah tabel berhasil diblok, langkah selanjutnya adalah mengklik kanan pada area yang telah dipilih. Pada *menu* yang muncul, cari dan pilih opsi yang bertuliskan "*View More Cell Actions*." Setelah itu, pilihlah opsi "*Conditional formatting*." Dengan memilih opsi ini, Anda akan dapat mengakses berbagai pengaturan format yang tersedia untuk sel.

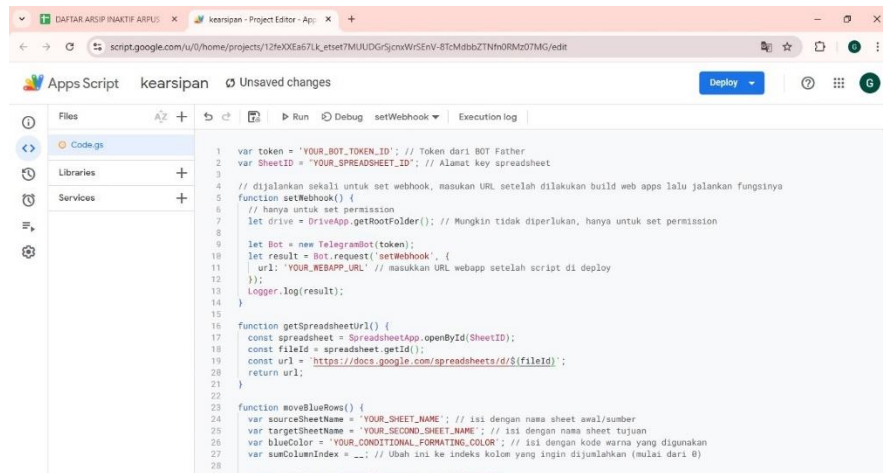
- 3) Kemudian, perhatikan bagian yang bertuliskan "*Format rules*." Anda perlu mengubah pengaturan di bagian ini. Secara default, pengaturan ini mungkin berada pada "*Is not empty*." Oleh karena itu, ubahlah menjadi "*Custom formula is*" untuk memberikan Anda lebih banyak fleksibilitas dalam menetapkan aturan format.
- 4) Selanjutnya, pada bagian "*Value or formula*," masukkan rumus yang telah disiapkan. Rumus yang dimaksud adalah ($=\$O10 \leq \text{TODAY}()$), di mana kolom O adalah lokasi tanggal habis retensi arsip untuk arsip bernomor satu. Pastikan Anda menempatkan tanda dolar di depan kolom untuk memastikan rumus ini tetap konsisten saat diterapkan ke seluruh tabel.
- 5) Setelah memasukkan rumus, langkah berikutnya adalah memilih warna untuk menandai arsip yang sudah habis masa retensinya. Anda dapat memilih warna default yang telah ditentukan, yaitu dengan kode warna (#b7e1cd). Warna ini akan membantu membedakan arsip yang telah mencapai batas waktu retensinya dengan mudah.
- 6) Terakhir, setelah semua pengaturan selesai dan Anda telah memilih warna yang diinginkan, klik tombol "*Done*."



Gambar 4. 7 Pemilihan Warna Conditional Formatting

4.2.7 Penginputan *Script* Pada *AppsScript*

Salah satu kelebihan dari *Spreadsheet* adalah terdapat banyak ekstensi tambahan yang memudahkan kita untuk melakukan pengolahan data secara otomatis hanya dengan memasukkan *Script* atau pemrograman di dalamnya. Hal ini tentu akan mempermudah dan menghemat waktu dalam pengerjaan data dengan jumlah yang banyak. Selain itu, kemampuan untuk mengintegrasikan berbagai *sumber* data juga memungkinkan pengguna untuk mengakses dan menganalisis informasi secara lebih komprehensif. Dengan demikian, pengguna dapat menghasilkan laporan yang lebih akurat dan tepat waktu, serta meningkatkan produktivitas secara keseluruhan. Ekstensi ini juga sering dilengkapi dengan fitur-fitur analisis lanjutan, yang memungkinkan pengguna untuk mendapatkan wawasan lebih dalam dari data yang mereka olah.



Berikut ini adalah langkah-langkah dalam penginputan dan pengeditan *Script* pada *SpreadSheet*:

- 1) Klik *menu Extensions* yang berada di *dashboard Spreadsheet*.
- 2) Selanjutnya pilih opsi *AppsScript*.
- 3) *Copy Script* lalu tempelkan *Script*. Setelah itu, Anda dapat memodifikasi *Script* sesuai kebutuhan dengan menambahkan fungsi tambahan atau mengubah parameter yang ada. Pastikan untuk menyimpan perubahan yang telah dilakukan dengan mengklik tombol *Save* di pojok kiri atas.

4.2.8 Pengeditan *Script*

Terdapat beberapa variabel di dalam *Script* yang perlu diubah supaya *Script* dapat berjalan dan tidak error. Proses pengeditan ini sangat penting untuk memastikan bahwa *Script* berfungsi dengan baik sesuai dengan kebutuhan. Berikut ini adalah tahapan-tahapan dalam pengeditan *Script* yang dapat diikuti secara sistematis:

1) **Masukkan *SpreadSheet ID* pada variabel nomor dua.**

SpreadSheet ID dapat dilihat pada halaman *SpreadSheet*, tepatnya berada pada *URLSpreadSheet*. Memastikan *ID* yang tepat sangat penting agar *Script* dapat terhubung dengan benar ke data yang dimaksud.

```
1 var token = 'YOUR_BOT_TOKEN_ID'; // Token dari BOT Father
2 var SheetID = "YOUR_SPREADSHEET_ID"; // Alamat key spreadsheet
```

Gambar 4. 9 Variabel Nomor 2

2) **Buat dua *Sheet* pada *SpreadSheet*.**

Sheet yang pertama berisi daftar arsip inaktif, sedangkan *Sheet* yang kedua nantinya akan berisi daftar arsip inaktif yang sudah bisa dimusnahkan, atau yang sudah habis masa retensinya. Dalam pemberian nama *Sheet*, pastikan setiap spasi menggunakan tanda “_”; hal ini bertujuan untuk memudahkan dalam pembacaan *Script*. Memperhatikan format penamaan ini juga dapat mencegah kesalahan yang dapat terjadi selama proses pengkodean. Pastikan format tabel yang berada pada *Sheet* kedua sama dengan *Sheet* yang pertama, sehingga konsistensi data terjaga.

6	050	Rencana Kerja dan Rancangan Akhir Rencana Kerja Dinas Kearsipan dan Perpustakaan Kabupaten Cilacap Tahun 2020	Juli	2018		2	R.023
+	≡	Daftar_arsip_Inaktif ▾	Arsip_Habis_Retensi ▾				

Gambar 4. 10 Format Penamaan *Sheet*

- 3) **Masukkan nama *Sheet* yang berisi daftar arsip inaktif pada variabel ke-24.**

Ini adalah langkah penting untuk memastikan *Script* dapat mengakses data dengan benar.

- 4) **Masukkan nama *Sheet* yang nantinya akan berisi daftar arsip yang sudah habis masa retensinya ke variabel 25.**

Penamaan yang konsisten akan memudahkan saat melakukan pemanggilan data.

- 5) **Masukkan kode warna yang digunakan pada saat tahapan pengaturan *Conditional formatting* ke variabel 26.**

Kode warna ini penting untuk visualisasi data, sehingga memudahkan pengguna dalam membaca status arsip.

- 6) **Pada variabel ke-27 masukkan *row* jumlah berkas.**

Row berbeda dengan kolom; jika kolom dimulai dari satu, maka *row* dimulai dari nol. Penjelasan ini penting agar pengguna tidak mengalami kebingungan saat mengatur data..

```
21 }
22
23 function moveBlueRows() {
24   var sourceSheetName = 'YOUR_SHEET_NAME'; // isi dengan nama sheet awal/sumber
25   var targetSheetName = 'YOUR_SECOND_SHEET_NAME'; // isi dengan nama sheet tujuan
26   var blueColor = 'YOUR_CONDITIONAL_FORMATTING_COLOR'; // isi dengan kode warna yang digunakan
27   var sumColumnIndex = __; // Ubah ini ke indeks kolom yang ingin dijumlahkan (mulai dari 0)
  ...
```

Gambar 4. 11 Variabel 24,25,26 dan 27

- 7) **Masukkan *row* Uraian Masalah pada variabel ke-62. *Row***

Uraian Masalah dimasukkan dengan tujuan untuk menampilkan arsip yang sudah habis masa retensinya pada pesan notifikasi di Telegram. Pengaturan ini sangat berguna untuk memantau status arsip secara *real-time*.

- 8) **Pada variabel ke-63, masukkan *row* jumlah berkas.**

Jumlah berkas nantinya akan dimunculkan di pesan

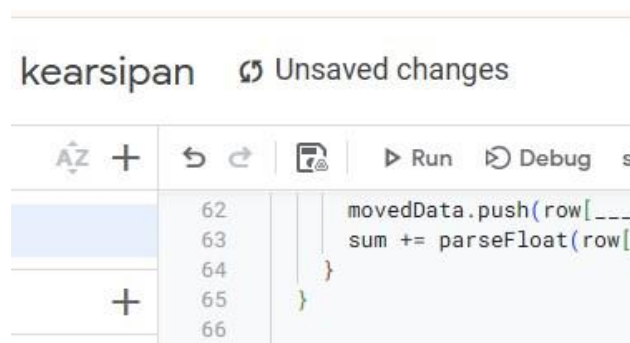
notifikasi, sehingga memudahkan pengguna untuk mengetahui angka pasti dari arsip yang perlu diperhatikan..

```
58  if (isBlueRow) {  
59      targetSheet.appendRow(row);  
60      rowsToDelete.push(i + 1);  
61      movedRows++;  
62      movedData.push(row[_____]); // Row yang menjadi acuan nama arsip yang akan ditampilkan di pesan notifikasi  
63      sum += parseFloat(row[_____]) || 0; // Row yang menjadi acuan jumlah berkas  
64  }  
65  }  
66
```

Gambar 4. 12 Variabel 62 dan 63

9) ***Save project*** supaya perubahan yang dilakukan pada ***Script*** dapat tersimpan.

Melakukan penyimpanan secara rutin adalah langkah yang bijak untuk mencegah kehilangan data atau perubahan yang sudah dilakukan. Pastikan pula untuk melakukan pengujian *Script* setelah semua perubahan selesai agar fungsionalitas dapat dipastikan dengan baik..



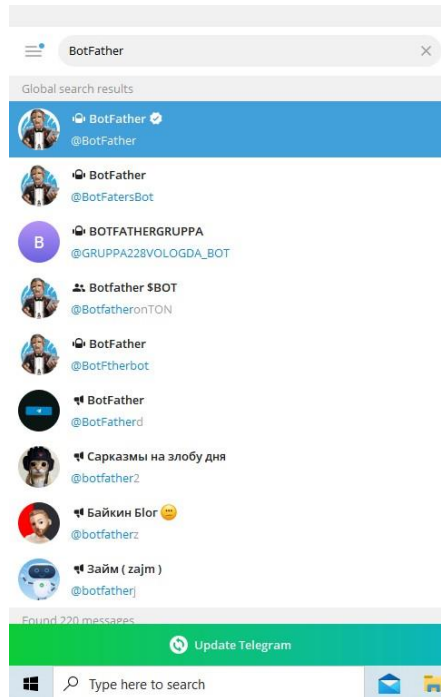
Gambar 4. 13 Save Project

4.2.9 Pembuatan *Bot* Telegram

Sebelum membuat *Bot* Telegram, pengguna diharuskan menginstal terlebih dahulu aplikasi Telegram. Proses penginstalan aplikasi ini sangat mudah dan bisa dilakukan lewat berbagai perangkat, baik itu smartphone maupun desktop. Untuk pengguna smartphone, aplikasi Telegram tersedia secara gratis di *Play store* untuk perangkat Android, sehingga pengguna hanya perlu mencarinya melalui kolom pencarian di aplikasi tersebut dan melakukan pengunduhan. Sementara itu, untuk pengguna desktop, aplikasi Telegram dapat diunduh melalui situs resmi Telegram di Google, di mana pengguna akan menemukan versi aplikasi yang sesuai dengan sistem operasi yang mereka gunakan, baik *Windows*, *macOS*, atau *Linux*.

Setelah menginstal aplikasi Telegram, pengguna perlu melakukan registrasi dan membuat akun jika belum memilikinya. Dengan akun yang sudah aktif, pengguna dapat langsung mulai membuat *bot* Telegram. Berikut ini tahapan dalam membuat *bot* Telegram:

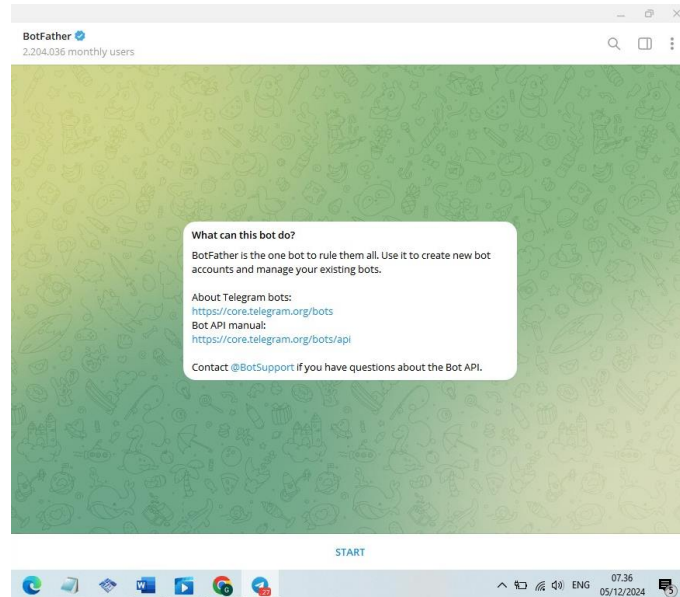
- 1) Membuka aplikasi Telegram dan pada bagian pencarian ketik “*Botfather*”. *Botfather* yang digunakan adalah yang sudah mendapatkan *verified* atau ceklist berwarna biru, menandakan bahwa *bot* tersebut resmi dan dapat diandalkan.



Gambar 4. 14 Tampilan Bot Father

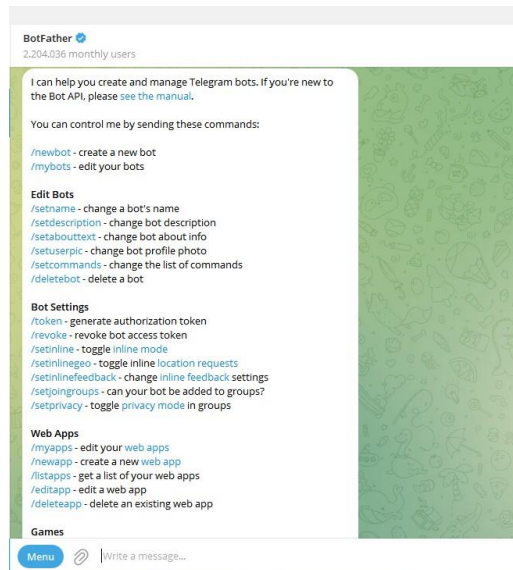
Pada gambar di atas terdapat banyak pilihan *Botfather*, namun yang digunakan dalam membuat *Cbot* Telegram merupakan yang terdapat ceklist-nya, atau tanda *verified*. Penting untuk memilih *Botfather* yang tepat agar segala proses pembuatan *bot* berjalan dengan lancar tanpa kendala.

- 2) Tekan *Start* pada *Botfather* untuk memulai membuat *bot* Telegram. Setelah menekan tombol *Start*, Anda akan melihat tampilan *menu* yang menyajikan berbagai opsi yang dapat dipilih. Ini adalah langkah awal yang krusial untuk memulai seluruh proses pembuatan *bot* dengan sempurna.



Gambar 4. 15 Tampilan Awal Bot Father

- 3) Setelah tombol *Start* ditekan, akan muncul beberapa pilihan *menu* pada *Botfather*. Untuk membuat *Bot* baru, pilih *menu newbot*. Ini adalah langkah penting yang akan membawa Anda kepada langkah-langkah berikutnya, di mana Anda mulai merancang dan menyesuaikan *bot* sesuai dengan kebutuhan Anda.



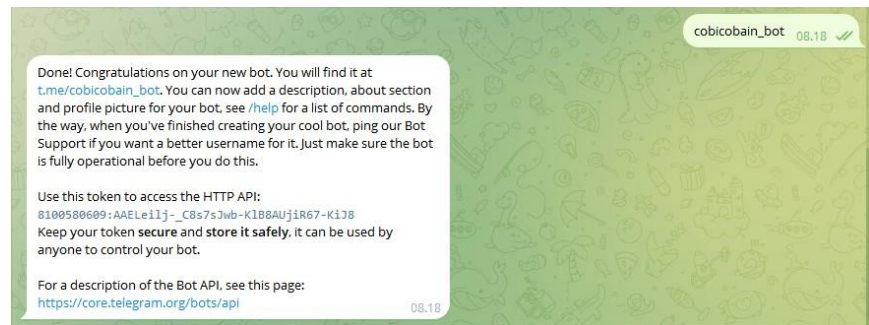
Gambar 4. 16 Tampilan Menu Awal Bot Father

- 4) Setelah *menu newbot* dipilih, pengguna diharuskan untuk memasukkan nama *bot* yang ingin mereka buat. Penamaan *Bot* sangat penting dan bebas berdasarkan keinginan dari pengguna. Nama *bot* yang unik dan menarik dapat menarik perhatian pengguna lain dan membuat *bot* Anda lebih dikenal di kalangan pengguna Telegram.



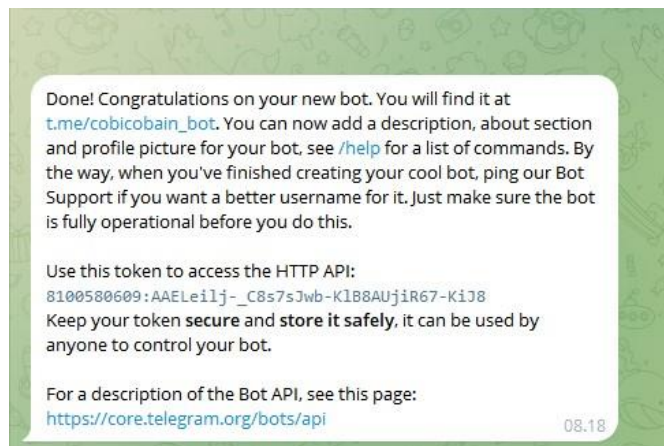
Gambar 4. 17 Input Nama Bot

- 5) Tahapan selanjutnya setelah nama untuk *Bot* ditentukan adalah menentukan *Username Bot*. *Username* berfungsi sebagai nama panggilan *bot*. Untuk penamaan *Username bot*, diharuskan menggunakan kata “*bot*” di akhir nama, dan penggunaan spasi diganti dengan tanda “*_*”.



Gambar 4. 18 Tampilan Berhasil Memasukan Username

Gambar di atas adalah contoh penamaan *bot* yang benar. *Username* dari *bot* tersebut adalah “cobicobain”. Sehingga *Username* botnya menjadi “cobicobain_bot”, karena diharuskan terdapat kata “bot” pada penamaan *Username bot*-nya. Selain itu, penggunaan spasi juga diganti dengan “_”.



Gambar 4. 19 Bot Token

Setelah memasukkan *Username* untuk *bot*, *Botfather* akan mengirimkan pesan yang berisi *Bot token*, yang digunakan untuk menghubungkan antara *bot* Telegram dengan *AppsScript*. *Bot token* ini adalah kunci akses Anda untuk melakukan berbagai interaksi dan pengaturan lebih lanjut.

- 6) Tahapan selanjutnya adalah memasukkan *Bot token* yang didapat dari *Botfather* ke *AppsScript*. Masukkan *Bot token*

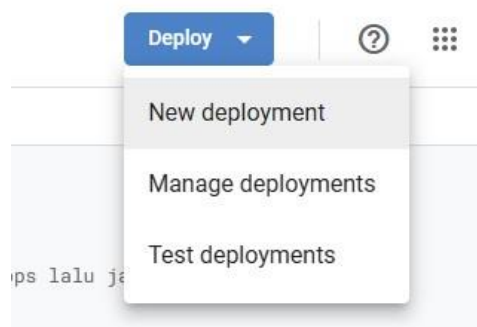
pada variabel ke-1. Keberadaan *Bot token* sangat penting karena tanpa token ini, *bot* Anda tidak akan memiliki koneksi yang diperlukan untuk berfungsi dengan baik.



```
1 var token = 'YOUR_BOT_TOKEN_ID'; // Token dari BOT Father
2 var SheetID = "YOUR_SPREADSHEET_ID"; // Alamat key spreadsheet
3
```

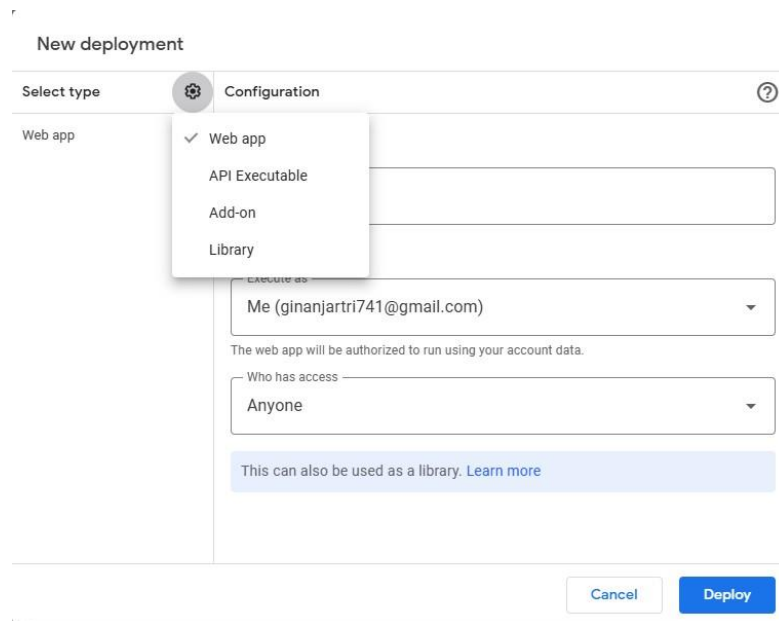
Gambar 4. 20 Variabel ke-1

- 7) Setelah memasukkan *Bot token* pada variabel satu, tahapan selanjutnya adalah *mendeploy Script*. Pada *menu deploy*, pilih opsi *new deployment* untuk memastikan bahwa semua perubahan yang telah dilakukan dapat diterapkan secara efektif. Proses ini penting untuk mengaktifkan *Script* Anda sehingga *bot* dapat berfungsi sesuai harapan.



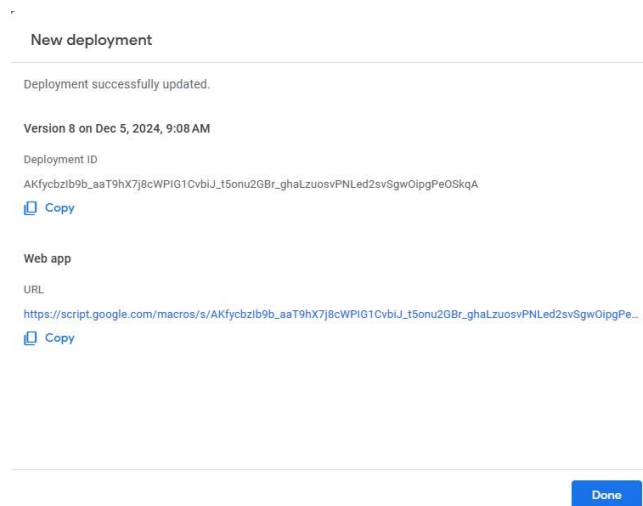
Gambar 4. 21 Tampilan Menu Deploy

- 8) Setelah *menu New deployment* dipilih, akan muncul tampilan dari *New deployment*. Ubah *Select type* menjadi *Webapp* dan kemudian tekan *Deploy*.



Gambar 4. 22 Tampilan New Deployment

Setelah menekan *Deploy*, akan muncul tampilan baru yang berisikan keterangan bahwa *deploy* berhasil dilakukan. Proses ini menandakan bahwa *bot* Anda kini sudah dapat diakses dan berfungsi.



Gambar 4. 23 Tampilan Setelah Berhasil Deploy Pada Menu New Deployment

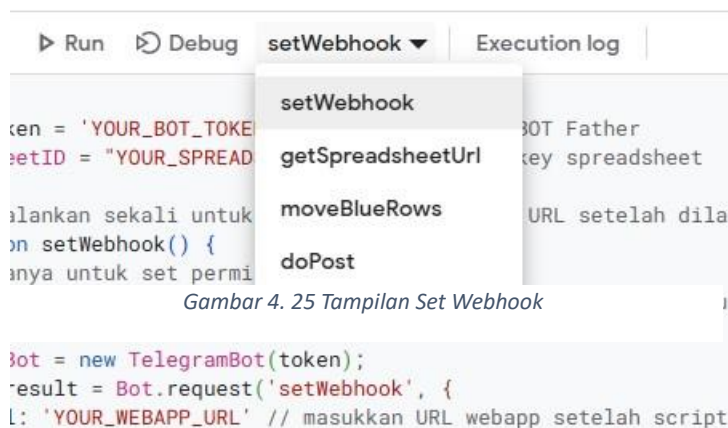
- 9) Tahapan selanjutnya adalah memasukkan *URLWeb app* pada variabel ke-11 di *AppsScript*. Memastikan bahwa *URL* ini

dimasukkan dengan benar sangat krusial, karena kesalahan sekecil apapun dapat mengakibatkan kegagalan dalam koneksi *bot* dengan *Script*.

```
9   let Bot = new TelegramBot(token);
10  let result = Bot.request('setWebhook', {
11    url: 'YOUR_WEBAPP_URL' // masukkan URL webapp setelah script di deploy
12  });
13  Logger.log(result);
14  }
15
```

Gambar 4. 24 Variabel Ke-11

10) Setelah *URLwebapp* dimasukkan, tahapan selanjutnya adalah menjalankan fungsi *setwebhook*. Fungsi *setwebhook* dijalankan supaya *bot* Telegram yang baru dibuat dapat terhubung dengan *Script* yang ada di *AppsScript*. Sehingga nantinya fungsi-fungsi yang ada di *Script* dapat dijalankan oleh *Bot* Telegram. Untuk menjalankan fungsi *setwebhook*, pilih opsi *setwebhook*, lalu tekan run untuk menjalankan fungsi tersebut.



Gambar 4. 25 Tampilan Set Webhook

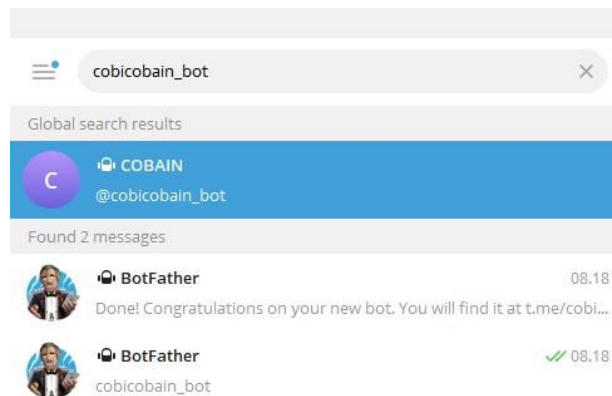
Fungsi *setwebhook* hanya diperlukan sekali saja untuk menghubungkan *bot* dan *Script*; menjalankannya berulang kali akan memutuskan koneksi yang telah terjalin.

Execution log		
11:02:35 AM	Notice	Execution started
11:02:45 AM	Info	Request URL: https://api.telegram.org/bot8100580609:AAELei
11:02:45 AM	Info	Request Options: {"method":"post","contentType":"applicati {\"url\": \"https://script.google.com/macros/s/AKfycbz0VnOf YzZSYhWWtWmZ5jaCLvCISRCHw/exec\"}"}
11:02:45 AM	Info	Response: {\"ok\":true,\"result\":true,\"description\":\"Webhook
11:02:45 AM	Info	{\"ok\":true,\"result\":true,\"description\":\"Webhook was set\"}
11:02:44 AM	Notice	Execution completed

Gambar 4. 26 Set Webhook Berhasil Dijalankan

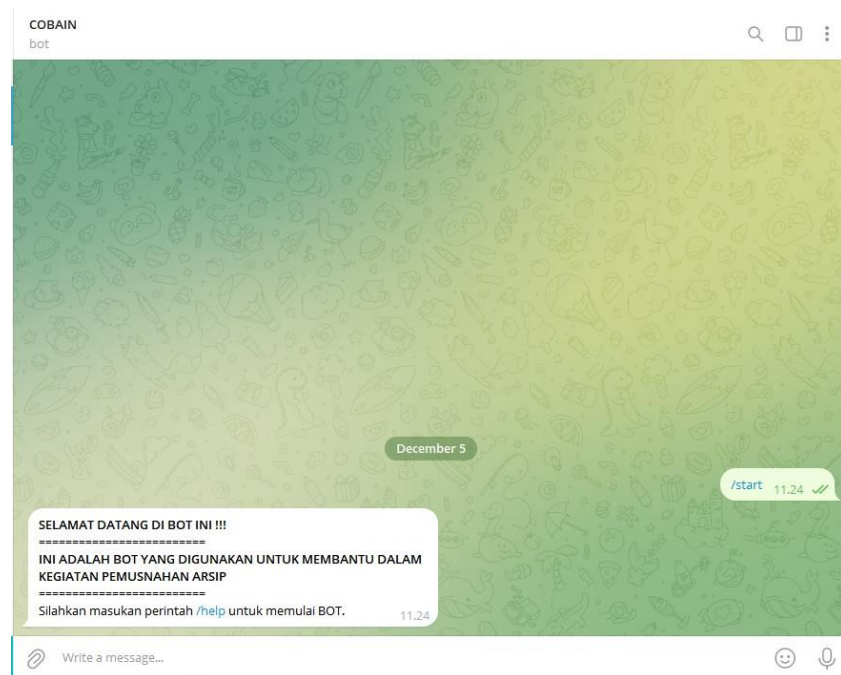
Gambar di atas adalah tampilan yang akan muncul apabila fungsi *setwebhook* berhasil untuk dijalankan.

- 11) Setelah berhasil menjalankan fungsi *setwebhook*, tahapan selanjutnya adalah mencari *bot* yang baru saja dibuat pada pencarian Telegram. Ini adalah langkah akhir yang menandakan bahwa Anda sudah berhasil membuat *Bot* Telegram dengan mengikuti prosedur yang tepat. Saat Anda menemukan *bot* tersebut, Anda dapat mulai berinteraksi dan mengonfigurasi *bot* sesuai kebutuhan Anda,



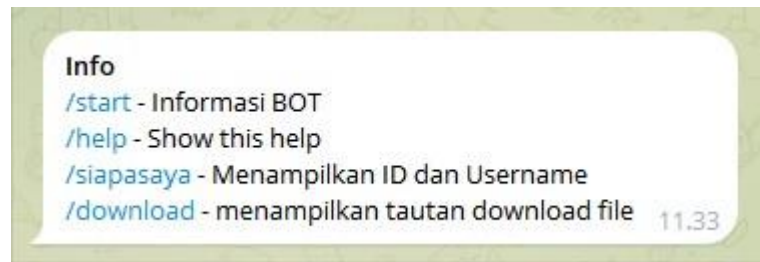
Gambar 4. 27 Tampilan Kolom Pencarian Telegram

12) Langkah berikutnya dalam proses ini adalah menemukan *Chat id* dari akun Telegram yang Anda miliki. Langkah pertama yang perlu dilakukan adalah menekan opsi "*Start*" pada *Bot* Telegram yang baru saja Anda buat. Setelah Anda melakukan hal tersebut, Anda akan diarahkan ke *menu* awal dari *bot*, di mana Anda bisa mulai menjelajahi berbagai opsi yang tersedia. Selanjutnya, untuk menemukan *Chat id* yang dibutuhkan, Anda perlu memilih opsi *"/help"* yang ada di dalam *menu Bot* tersebut. Ketika opsi ini ditekan, Anda akan melihat tampilan instruksi yang memberikan petunjuk lengkap tentang cara penggunaan *bot* secara efisien. Dalam daftar opsi yang muncul, carilah dan pilih opsi *"/siapasaya"* yang akan menampilkan informasi penting seperti *Chat id* serta *Username* akun Telegram Anda.



Gambar 4. 28 Tampilan Awal Bot Setelah di Start

Untuk mencari *Chat id*, langkah pertama yang harus dilakukan adalah memilih opsi `"/help"` yang tersedia di dalam *Bot*. Setelah Anda menekan opsi ini, akan muncul instruksi yang jelas dan terperinci mengenai cara penggunaan *bot*, sehingga dapat dipahami semua fungsionalitas yang ditawarkan. Instruksi tersebut biasanya mencakup panduan langkah demi langkah yang membantu mengenali berbagai perintah dan fitur yang dapat digunakan. Dengan begitu, tidak hanya dapat menemukan *Chat id*, tetapi juga mempelajari cara memaksimalkan pengalaman menggunakan *bot* tersebut. Hal ini sangat berguna terutama bagi pengguna baru yang mungkin belum akrab dengan cara kerja *bot* Telegram. Selain itu, dengan memahami instruksi yang diberikan, Pengguna dapat lebih percaya diri dalam menjelajahi opsi lainnya yang ada di dalam *Bot*, sehingga proses interaksi menjadi lebih lancar dan efisien.



Gambar 4. 29 List Perintah Pada Bot

13) Tahapan berikutnya adalah memasukkan *Chat id* ke variabel nomor 194 pada *Apps Script*. Proses ini penting karena variabel ini akan berfungsi sebagai penghubung antara *bot* dan akun Telegram pengguna, sehingga semua pesan yang dikirim dapat terarah dengan tepat..

```

191
192   sendMessage(text, parse_mode = 'HTML') {
193     let data = {
194       chat_id: TelegramData ? TelegramData.message.chat.id : 'YOUR_CHAT_ID', // Ganti
195       text: text,
196       parse_mode: parse_mode
197     };
198     this.request('sendMessage', data);
199   }
200

```

Gambar 4. 30 Variabel ke-194

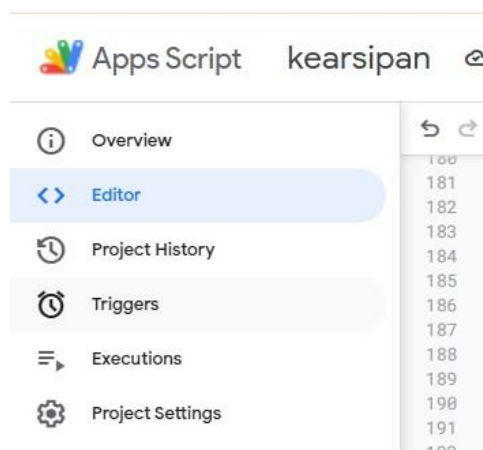
Setelah *Chat id* dimasukkan ke dalam variabel ke-194, langkah selanjutnya adalah melakukan proses pengaturan ulang atau *deploy*, sama seperti yang pengguna lakukan pada langkah sebelumnya. Proses *deployment* ini memastikan bahwa semua perubahan yang telah pengguna buat dalam *Script* dapat diterapkan dan berfungsi dengan baik. Penting untuk dicatat bahwa setiap kali terjadi perubahan pada *Script*, pengguna diharuskan untuk melakukan *new deployment* lagi agar setiap pembaruan dapat diimplementasikan. Dengan mengikuti langkah-langkah ini secara seksama, Pengguna akan memastikan bahwa *bot* Telegram yang pengguna buat berfungsi sesuai dengan yang diharapkan. Dengan demikian,

tahapan pembuatan *bot* sudah selesai, dan pengguna kini siap untuk mulai menggunakan *bot* tersebut serta mengeksplorasi berbagai kemungkinannya yang sangat bermanfaat..

4.2.10 Pemasangan Trigger Pada Apps Script

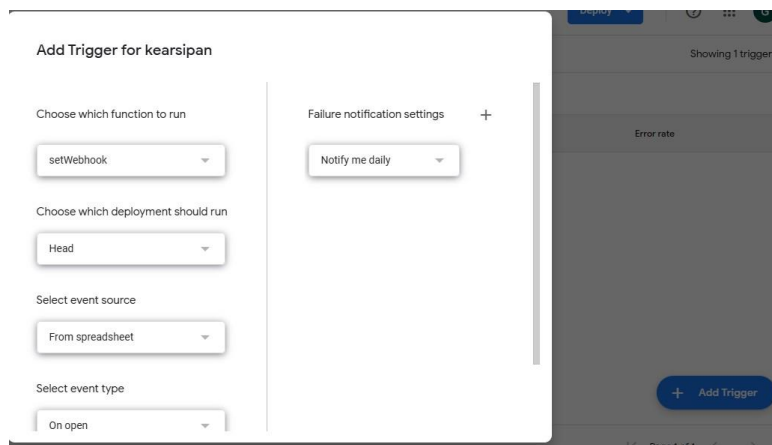
Pemasangan Trigger pada Apps Script merupakan langkah penting yang memungkinkan program untuk beroperasi secara otomatis sesuai dengan kondisi yang telah ditentukan. Trigger berfungsi untuk menjalankan fungsi secara otomatis ketika peristiwa tertentu terjadi, seperti misalnya setiap tanggal satu pada awal bulan, ketika laporan bulanan perlu dikirimkan. Dengan adanya trigger, tidak perlu lagi melakukan proses manual, sehingga efisiensi waktu dan tenaga dapat dijaga. Berikut ini adalah beberapa tahapan dalam pemasangan Trigger pada AppScript :

- 1) Pilih opsi Triggers pada *menu Apps Script* untuk mengakses pengaturan yang berkaitan dengan fungsi otomatisasi. *Menu* ini memberikan gambaran umum tentang semua trigger yang telah diatur sebelumnya, serta memungkinkan pengguna untuk menambah atau mengedit trigger yang diinginkan.



Gambar 4. 31 Menu AppsScript

- 2) Pilih opsi *Add trigger* yang berada pada bagian pojok kanan bawah untuk memulai proses penambahan trigger baru. Setelah opsi ini dipilih, berbagai pilihan konfigurasi akan muncul, memungkinkan pengguna untuk menentukan fungsi mana yang ingin dijalankan secara otomatis. Pada tahap ini, pengguna perlu mengatur jenis event yang akan memicu fungsi tersebut, seperti waktu tertentu atau perubahan pada dokumen. Proses ini dapat diulang untuk menambahkan beberapa trigger sesuai kebutuhan, sehingga program dapat beroperasi dengan lebih efisien dan sesuai dengan jadwal yang diinginkan.

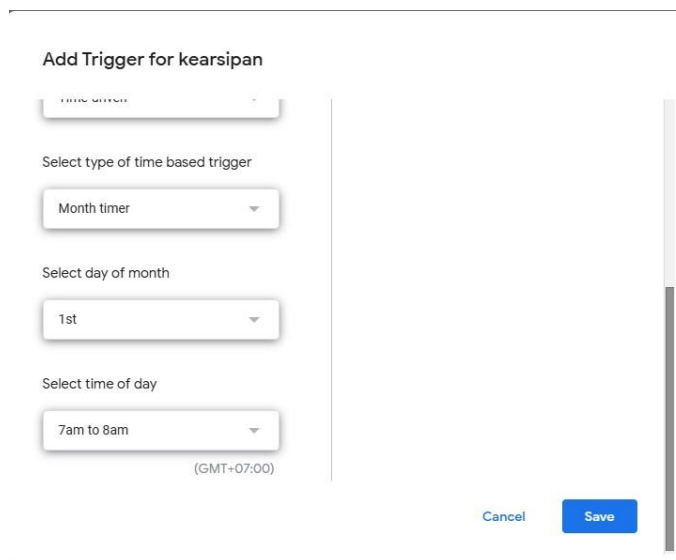


Gambar 4. 32 Tampilan menu Add Trigger

- 3) Pada bagian *Choose which function to run*, pilih *movebluerows*. Pemilihan fungsi ini sangat penting karena fungsi yang akan dijalankan hanyalah *movebluerows*, yang dirancang khusus untuk mengelola data arsip. Fungsi ini bertujuan untuk menangani arsip yang sudah habis masa retensinya dengan cara yang efisien dan terstruktur. Dengan demikian, fungsi ini akan secara otomatis memindahkan arsip tersebut ke *Sheet* kedua, yang telah disiapkan untuk menyimpan daftar arsip yang telah melewati masa

retensinya. *Sheet* kedua ini berfungsi sebagai tempat pemantauan dan pengelolaan arsip yang telah kedaluwarsa, membuatnya lebih mudah untuk diakses dan dikelola. Dalam proses ini, *movebluerows* tidak hanya melakukan pemindahan data, tetapi juga membantu menjaga keteraturan dan kerapian dalam pengelolaan arsip, memastikan bahwa data yang relevan tetap berada di tempat yang tepat.

- 4) Selanjutnya, pada bagian *select event source*, pilih *time driven*. Pilihan *time driven* diambil karena fungsi yang diinginkan hanya akan berjalan berdasarkan perhitungan waktu yang telah ditentukan sebelumnya. Artinya, skrip akan aktif dan menjalankan tugasnya hanya ketika waktu yang telah dijadwalkan tiba. Konsep ini menyerupai fungsi alarm, di mana alarm akan berbunyi dan mengingatkan pada waktu tertentu; demikian pula, skrip ini akan memulai eksekusinya pada waktu yang sudah ditetapkan. Dengan pendekatan ini, pengguna dapat mengatur skrip untuk melakukan berbagai tugas secara otomatis, seperti menghapus atau memindahkan data, tanpa perlu campur tangan manual.



Add Trigger for kearsipan

TIME SOURCE

Select type of time based trigger

Month timer

Select day of month

1st

Select time of day

7am to 8am

(GMT+07:00)

Cancel Save

Gambar 4. 33 Tampilan Menu Add Trigger ke-2

- 5) Pada bagian *select type of time based trigger*, pilih *month time*. Pemilihan *month time* dilakukan dengan pertimbangan bahwa skrip ini akan dijadwalkan untuk berjalan setiap satu bulan sekali, memungkinkan setiap proses yang dijalankan untuk bersifat berkala dan teratur. Dengan menggunakan pengaturan ini, pengguna dapat memastikan bahwa aktivitas tertentu, seperti pengarsipan atau pembersihan data, dilakukan secara konsisten tanpa perlu pengaturan ulang setiap kali. Keputusan untuk memilih *trigger* bulanan sangat strategis, terutama untuk tugas-tugas yang tidak memerlukan frekuensi lebih tinggi, sehingga menghemat *sumber* daya dan waktu. Selain itu, dengan penjadwalan bulanan, pengguna dapat merencanakan momentum penting dalam pengelolaan data, memungkinkan tinjauan berkala terhadap arsip dan memastikan bahwa semua informasi yang sensitif atau penting dikelola dengan baik.
- 6) Selanjutnya, pada bagian *select day of month*, pilih 1st. Hal ini berarti skrip akan berjalan satu bulan sekali setiap tanggal 1, yang merupakan waktu yang strategis untuk memeriksa dan mengelola arsip. Pilihan ini diambil karena semua arsip yang diatur memiliki tanggal habis retensi yang rata-rata jatuh pada tanggal 1, sehingga membuat pengelolaan dan pemantauan lebih efisien. Dengan pengaturan ini, setiap tanggal satu skrip akan melakukan pemeriksaan terhadap arsip yang telah melewati masa retensinya dan secara otomatis akan mengirimkan notifikasi kepada pihak terkait mengenai arsip yang memerlukan tindakan lebih lanjut. Dengan cara ini, pengguna dapat secara proaktif menangani arsip yang kedaluwarsa,

- 7) Pada bagian memilih waktu, disarankan untuk memilih rentang 7 pagi hingga 8 pagi. Pilihan ini diambil karena jam kerja pegawai kantor dimulai pada pukul 7 pagi. Dengan demikian, waktu tersebut adalah saat di mana aktivitas kerja biasanya berlangsung. Oleh karena itu, menetapkan waktu antara 7 hingga 8 pagi menjadi keputusan yang tepat dan relevan untuk mencocokkan dengan jadwal kerja yang ada.



Gambar 4. 34 Tampilan Notifikasi Yang Dikirimkan bot Telegram

Gambar di atas merupakan ilustrasi notifikasi yang dikirimkan oleh *bot* Telegram terkait arsip yang telah melewati masa retensinya. Dalam notifikasi tersebut, terdapat informasi penting mengenai nama arsip yang sudah tidak lagi memenuhi kriteria penyimpanan. Selanjutnya, keterangan mengenai jumlah total berkas yang sudah habis masa retensinya ditampilkan dengan jelas, memberikan gambaran yang komprehensif tentang kondisi arsip tersebut. Selain itu, pada bagian akhir notifikasi, disediakan opsi untuk mengunduh berkas melalui perintah */download*, yang memudahkan pengguna untuk mengambil tindakan sesuai kebutuhan mereka. Notifikasi semacam ini sangat penting untuk menjaga keteraturan dan efisiensi dalam pengelolaan arsip.

4.2.11 Analisis Perbandingan

Berdasarkan hasil penelitian, terdapat perbedaan signifikan antara metode manual dan otomatisasi dalam proses pencarian dan penentuan arsip yang telah habis masa retensinya. Pada metode manual, petugas harus membuka file Excel dan menelusuri satu per satu daftar arsip untuk menentukan mana yang telah habis masa retensinya. Proses ini memakan waktu rata-rata **3 jam per file**, tergantung pada jumlah arsip yang harus diperiksa. Selain memakan waktu, metode manual juga rentan terhadap kesalahan manusia, seperti tidak telitian dalam memeriksa tanggal retensi atau kelalaian dalam mencatat arsip yang telah habis masa retensinya.

Di sisi lain, metode otomatisasi yang dikembangkan dalam penelitian ini menggunakan *bot* Telegram yang terhubung dengan Google Spreadsheet. Sistem ini mampu melakukan pencarian dan penentuan arsip yang telah habis masa retensinya dalam waktu **2-3 detik**. Kecepatan ini dicapai melalui penggunaan *script* yang secara otomatis memindai data di Spreadsheet dan mengirimkan notifikasi kepada petugas. Selain itu, sistem ini juga mengurangi risiko kesalahan manusia karena proses pencarian dan penentuan arsip dilakukan secara terprogram.

Dari segi efisiensi waktu, metode otomatisasi menunjukkan peningkatan yang sangat signifikan. Jika metode manual membutuhkan **3 jam (10.800 detik)** untuk menyelesaikan satu file, sedangkan metode otomatisasi hanya membutuhkan **2-3 detik**, maka efisiensi waktu yang dicapai adalah sebesar **99,97%**. Perhitungan ini didasarkan pada rumus:

$$\text{Efisiensi} = 1 - \left(\frac{\text{Waktu Otomatisasi}}{\text{Waktu Manual}} \right) \times 100\%$$

$$\text{Efisiensi} = 1 - \left(\frac{3}{10.800} \right) \times 100 = 99,97\%$$

Executions

Showing 2 executions over last 7 days

Trigger ID: "QWZIJgxPSxuDUsxHeRQTeg" ✕

Deployment	Function	Type	Start Time	Duration	Status
Head	moveBlueRows	Trigger	Feb 19, 2025, 3:28:15 PM	2.117 s	Completed
Head	moveBlueRows	Trigger	Feb 18, 2025, 8:58:21 PM	3.703 s	Completed

Gambar 4. 35 Kecepatan Triggers

Selain efisiensi waktu, metode otomatisasi juga memberikan keuntungan lain, seperti mengurangi beban kerja petugas dan meningkatkan akurasi dalam menentukan arsip yang telah habis masa retensinya. Dengan demikian, dapat disimpulkan bahwa metode otomatisasi yang dikembangkan dalam penelitian ini tidak hanya lebih cepat, tetapi juga lebih andal dibandingkan dengan metode manual.

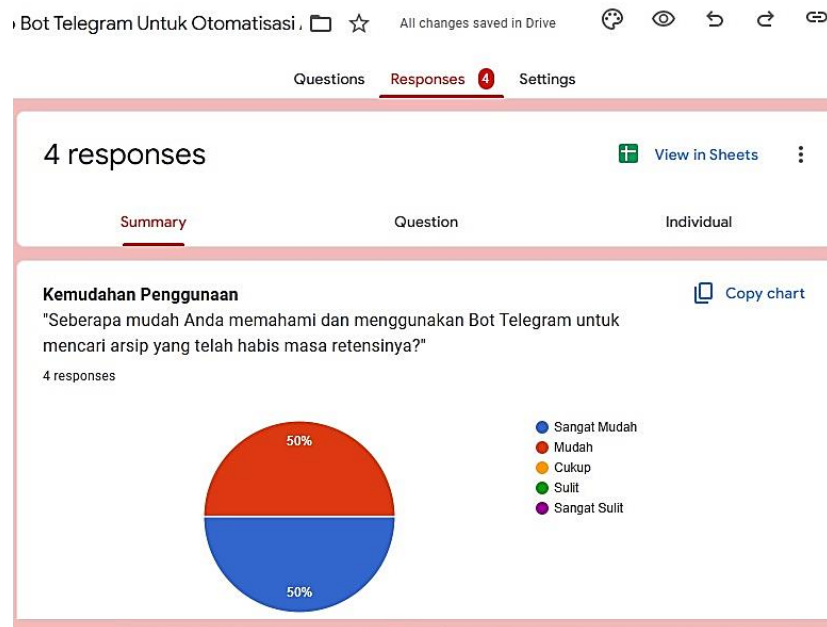
4.3 Sosialisasi dan Mekanisme Penggunaan

4.3.1 Sosialisasi

Setelah pengembangan Bot Telegram selesai, langkah selanjutnya adalah melakukan sosialisasi kepada pengguna, yaitu petugas pemusnahan arsip di Dinas Kearsipan dan Perpustakaan Kabupaten Cilacap. Sosialisasi ini bertujuan untuk memperkenalkan fitur-fitur Bot Telegram serta memberikan panduan penggunaan sistem otomatisasi yang telah dibuat. Mengingat keterbatasan pertemuan tatap muka, sosialisasi dilakukan secara online dengan mengirim pesan whatsapp yang berisi panduan dan tahapan-tahapan dalam penggunaan *bot* Telegram.

Selain itu Sebuah kuesioner online dikirimkan kepada petugas untuk mengumpulkan tanggapan mereka terkait

penggunaan Bot Telegram. Kuesioner ini berisi pertanyaan tentang kemudahan penggunaan, kecepatan notifikasi, dan saran perbaikan.



Gambar 4. 36 Survey Kemudahan Pengguna

Berdasarkan survey kemudahan pengguna. 50% menyatakan bahwa sistem sangat mudah digunakan dan 50% lagi menyatakan bot ini mudah digunakan. Hal ini bisa terjadi karena memang terdapat petugas yang sangat mengerti akan teknologi dan terdapat petugas yang hanya mengerti teknologi namun tidak terlalu mendalaminya. Berdasarkan survey ini bisa dikatakan sistem mudah untuk digunakan.

Kecepatan Notifikasi

"Seberapa cepat Bot Telegram memberikan notifikasi terkait arsip yang telah habis masa retensinya?"

4 responses

notifikasi biasa dikirimkan jam 7-8 pagi saat waktu kerja. Menurut saya itu waktu yang tepat dan cepat dalam mengirimkan notifikasi

bot ini memberikan kecepatan dalam pengiriman notifikasi. Notifikasi dikirim pada saat jam kerja antara jam 8-9. Notifikasi ini membantu karena dikirimkan pada saat jam kerja, tidak dikirimkan pada saat jam selesai kerja. Tentu saja ini memudahkan

Untuk cepatnya saya kurang paham, karena saya jarang memegang hp, tapi tiap kali saya memegang hp untuk istirahat biasanya ada notifikasi terkait arsip yang sudah habis masa retensinya

Pengiriman notifikasi relatif cukup cepat, karena jika ada arsip yang habis retensinya pada hari itu, maka pada hari itu juga kita akan menerima notifikasi dari bot telegram

Gambar 4. 37 Survey Kecepatan Notifikasi

Berdasarkan survey kecepatan notifikasi pengguna mengakui bahwa notifikasi yang dikirimkan tepat waktu dan juga tidak mengganggu aktivitas mereka ketika tidak sedang bekerja. Karena notifikasi hanya dikirimkan pada saat jam kerja yaitu antara jam 7-9 pagi.

Keandalan Sistem

"Seberapa andal Bot Telegram dalam memberikan informasi yang akurat tentang arsip yang telah habis masa retensinya?"

4 responses

jika dilihat dari retensi arsipnya, arsip yang dinyatakan habis retensi memang arsip yang sudah seharusnya habis retensi, cukup tepat

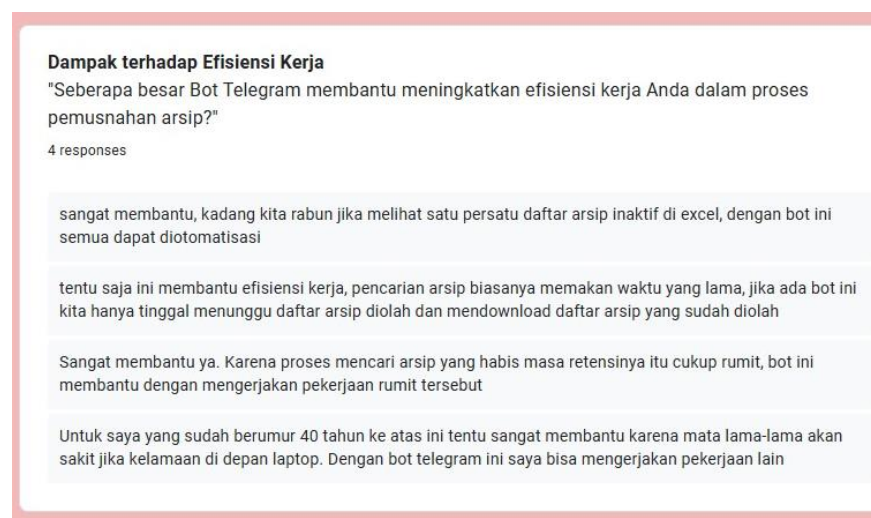
bot ini hanya mengolah arsip yang sudah habis masa retensinya, namun dalam kasus apakah arsip ini musnah atau tidak itu tetap menjadi kewenangan kami sebagai arsiparis dan pencipta arsip

Informasi arsip yang diberikan memang benar bahwa status arsip tersebut sudah habis retensi, namun saya tetap cross check lagi dan memang benar akurat

Informasi yang dikirimkan sesuai dengan nama arsip yang habis masa retensinya

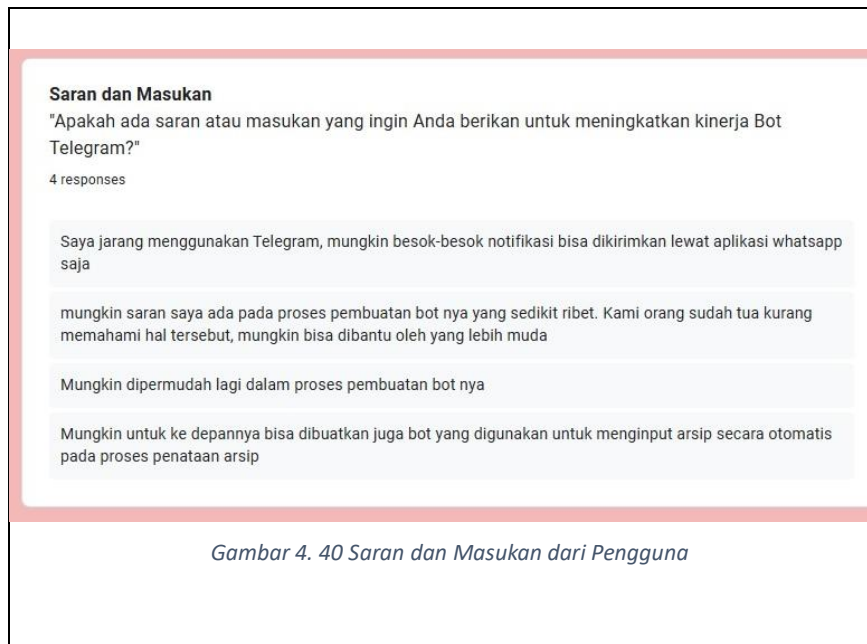
Gambar 4. 38 Survey Keandalan Sistem

Berdasarkan survey kepada pengguna sistem dapat diandalkan karena mengirimkan informasi yang tepat berupa daftar arsip yang sudah habis masa retensinya, namun ini hanya berupa daftar arsip yang sudah habis masa retensinya yang sesuai berdasarkan JRA. Untuk menentukan arsip tersebut akan musnah atau tidak itu tetap akan menjadi Keputusan arsiparis dan juga pencipta arsip. Sistem ini hanya memudahkan untuk pencarian arsip yang sudah habis masa retensinya.



Gambar 4. 39 Survey Efisiensi Kerja

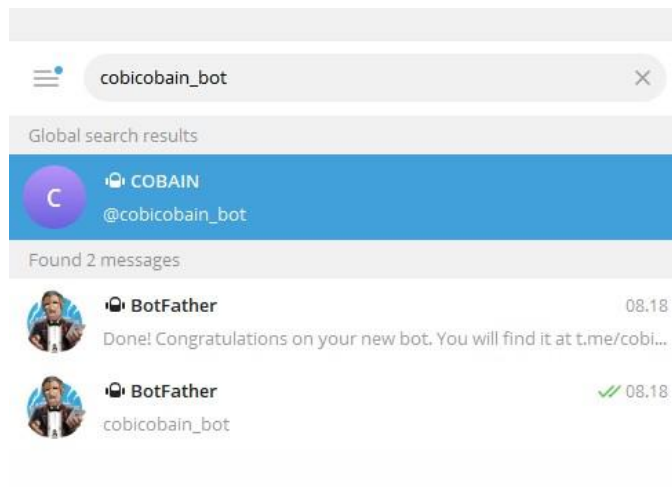
Berdasarkan survey terhadap pengguna, sistem mampu menghemat efisiensi pengguna yang cukup signifikan. Proses pencarian arsip biasa memakan waktu yang lama antara 2-3 jam, dengan adanya sistem ini pekerjaan petugas mampu dipangkas dan petugas dapat mengerjakan pekerjaan yang lain.



Berdasarkan saran dari pengguna, kebanyakan pengguna menyarankan untuk proses pembuatan *bot* Telegram dapat dipermudah lagi. Hal ini dikarenakan proses pembuatan *bot* Telegram membutuhkan waktu yang lama dan juga proses yang rumit. Kebanyakan pengguna menginginkan proses yang minimal namun mendapatkan hasil yang maksimal.

4.3.2 Daftar Perintah Pada *Bot*

Sebelum pengguna dapat memanfaatkan berbagai perintah yang tersedia, langkah awal yang perlu dilakukan adalah mengakses *bot* Telegram. Proses ini sangat penting agar pengguna dapat menjelajahi semua fitur yang ditawarkan. Untuk menemukan *bot*, pengguna cukup mengetikkan nama pengguna (*username*) *bot* yang spesifik pada kolom pencarian Telegram. Setelah *bot* ditemukan, pengguna hanya perlu menekan profil *bot* tersebut untuk memulai interaksi. Di dalam antarmuka *bot*, akan tersedia berbagai opsi dan perintah yang dapat dijelajahi lebih lanjut.

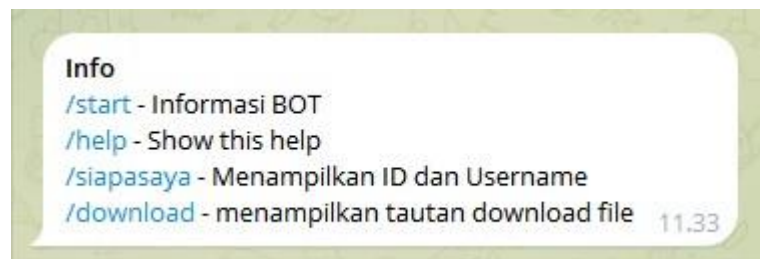


Gambar 4. 41 Kolom Pencarian Pada Telegram

Gambar di atas menampilkan kolom pencarian yang terdapat pada aplikasi Telegram, yang merupakan salah satu fitur penting dalam menemukan berbagai jenis konten dan *bot* dengan mudah. Setelah pengguna mengetikkan nama pengguna (*username*) *bot* yang spesifik pada kolom tersebut dan menekan tombol pencarian, hasil pencarian yang relevan akan ditampilkan secara instan. Proses ini memungkinkan pengguna untuk dengan cepat mengidentifikasi *bot* yang diinginkan tanpa harus menggulir melalui daftar yang panjang. Selain itu, hasil pencarian juga dapat mencakup berbagai *bot* dengan nama yang mirip, sehingga pengguna perlu memperhatikan detail lain, seperti gambar profil atau deskripsi singkat yang menyertainya, untuk memastikan bahwa pilihan yang diambil adalah *bot* yang tepat. Setelah hasilnya muncul, pengguna hanya perlu menekan pada nama *bot* untuk melanjutkan ke interaksi berikutnya.

Pengguna kemudian dapat memilih *bot* yang sesuai berdasarkan hasil pencarian dan mulai menjalin interaksi dengan menekan tombol '*Start*'. Tindakan ini sangat krusial karena akan segera memicu *bot* untuk menampilkan daftar perintah atau *menu* utama yang dapat digunakan. Setelah *menu* muncul, pengguna akan diberikan berbagai opsi yang disajikan dengan ringkas dan jelas, sehingga memudahkan dalam memilih tindakan selanjutnya. Setiap perintah yang tersedia dirancang untuk memfasilitasi pengguna dalam mengakses fungsi tertentu.

Setelah pengguna menekan tombol '*Start*', *bot* akan merespons dengan mengirimkan sebuah pesan notifikasi yang berisi informasi umum mengenai fungsionalitas dan tujuan dari *bot* tersebut. Informasi ini dapat berupa penjelasan singkat tentang fitur-fitur utama yang ditawarkan oleh *bot* dan cara penggunaannya. Pesan notifikasi ini dirancang untuk memberikan gambaran yang jelas mengenai manfaat utama yang bisa diperoleh, sehingga pengguna dapat segera memahami nilai tambah yang ditawarkan. Selain itu, *bot* juga akan menyertakan perintah '*/help*' sebagai panduan yang sangat berguna, memberikan akses cepat ke daftar lengkap perintah yang dapat diakses selama percakapan. Pengguna diberikan opsi untuk menggali lebih dalam tentang setiap fitur, sehingga membantu dalam memaksimalkan pengalaman penggunaan.



Gambar 4. 42 Daftar Perintah Pada Bot

Gambar di atas menampilkan daftar perintah yang dapat diakses pengguna setelah mengetikkan `"/help"`. Daftar ini memberikan kesempatan bagi pengguna untuk mengeksplorasi berbagai opsi yang tersedia, memastikan bahwa mereka dapat memanfaatkan setiap fungsionalitas yang ditawarkan oleh *bot*. Berikut penjelasan lengkap mengenai masing-masing perintah:

- 1) **/Start:** Perintah ini berfungsi sebagai pintu masuk utama untuk berinteraksi dengan *bot*. Saat pengguna mengetikkan `"/start"`, *bot* akan mengirimkan pesan selamat datang yang berisi informasi umum mengenai tujuan dan fungsionalitas *bot*. Pesan ini biasanya mencakup panduan singkat tentang cara menggunakan *bot* dan fitur-fitur utama yang tersedia. Selain itu, pesan selamat datang ini sering kali menyoroti manfaat utama yang dapat diperoleh dari penggunaan *bot*, seperti kemudahan akses informasi dan efisiensi dalam menyelesaikan tugas. Setelah menerima pesan tersebut, pengguna akan merasakan pengalaman yang lebih terarah dan dapat memahami dengan jelas langkah-langkah yang perlu diambil untuk memaksimalkan interaksi. Informasi lebih lanjut juga bisa mencakup rekomendasi perintah yang dapat dicoba segera, membantu pengguna untuk menavigasi sistem dengan lebih lancar.
- 2) **/help:** Perintah ini digunakan untuk menampilkan daftar lengkap perintah yang dapat digunakan oleh pengguna. Daftar perintah ini berfungsi sebagai panduan bagi pengguna untuk mengetahui tindakan apa saja yang dapat dilakukan dengan *bot*. Selain itu, daftar tersebut biasanya dilengkapi dengan deskripsi singkat mengenai setiap perintah, sehingga memudahkan pengguna untuk memahami fungsi spesifik masing-masing. Dengan adanya informasi ini, pengguna dapat dengan cepat menemukan perintah yang sesuai dengan kebutuhan mereka, baik itu untuk mendapatkan informasi, mengelola tugas, atau berinteraksi dalam

cara lain yang ditawarkan oleh *bot*. Selain itu, penyajian daftar perintah yang terstruktur dan mudah diakses membuat pengalaman pengguna semakin intuitif, mengurangi kebingungan yang mungkin timbul saat mencoba berbagai fitur. Pengguna juga sering kali diberi opsi untuk mencari perintah tertentu, yang lebih lanjut meningkatkan efisiensi dalam menemukan tindakan yang diinginkan. Dengan demikian, perintah ini tidak hanya berfungsi sebagai alat navigasi, tetapi juga sebagai *sumber* daya penting yang dapat mengedukasi pengguna tentang potensi penuh dari fitur-fitur yang tersedia, mendorong eksperimen dan eksplorasi yang lebih dalam.

- 3) **/siapasaya:** Perintah ini memberikan informasi identitas pengguna kepada *bot*. Ketika pengguna mengetikkan *"/siapasaya"*, *bot* akan merespons dengan mengirimkan pesan yang berisi *chat id* Telegram pengguna dan *username* mereka. *Chat id* ini memiliki peran penting dalam mekanisme notifikasi, terutama untuk mengirimkan pemberitahuan mengenai arsip yang sudah habis masa retensinya. Selain itu, informasi ini juga memungkinkan *bot* untuk mengelola interaksi secara lebih personal, memastikan bahwa setiap notifikasi yang dikirimkan relevan dan sesuai dengan kebutuhan pengguna. Dengan cara ini, pengguna dapat dengan mudah menerima informasi penting dan pembaruan yang berkaitan dengan aktivitas mereka. Keberadaan fitur ini diharapkan dapat meningkatkan pengalaman pengguna, membuat komunikasi menjadi lebih efisien dan terorganisir. Melalui penggunaan *chat id* dan *username*, *bot* dapat mengenali setiap pengguna secara unik, yang memungkinkan penyesuaian layanan sesuai dengan preferensi individu.
- 4) **/download:** Perintah ini memungkinkan pengguna untuk mengunduh dokumen yang berisi daftar arsip yang sudah habis masa retensinya. Setelah mengetikkan *"/download"*, pengguna

akan diarahkan ke sebuah halaman *spreadSheet* yang berisi informasi lengkap mengenai arsip-arsip tersebut. Untuk mengunduh dokumen, pengguna perlu mengakses menu "*File*" pada *spreadSheet* dan memilih opsi "*Download*." Proses ini tidak hanya memberikan kemudahan bagi pengguna dalam mengelola berkas, tetapi juga memastikan bahwa informasi terkait arsip dikelola secara sistematis dan transparan. Dalam *spreadSheet* tersebut, setiap arsip dilengkapi dengan detail penting, seperti tanggal kedaluwarsa, kategori, dan status pemusnahan, sehingga pengguna dapat dengan cepat mengevaluasi tindakan yang perlu diambil. Selain itu, dokumen yang diunduh dapat disimpan dan diarsipkan secara lokal untuk referensi di masa mendatang, memberikan fleksibilitas lebih dalam pengelolaan arsip. Dengan fitur ini, proses pemantauan dan pengelolaan arsip menjadi lebih efisien. Selanjutnya, pengguna dapat memilih format dokumen yang diinginkan (misalnya, PDF, Excel) untuk diunduh.

4.3.3 Mekanisme Pengiriman Notifikasi

Bot akan secara otomatis mengirimkan notifikasi kepada pengguna yang telah menentukan *chat id* mereka melalui perintah */siapasaya*. Notifikasi ini akan berisi informasi spesifik mengenai arsip yang telah melewati masa retensi. Isi notifikasi akan mencakup:

- Nama arsip: Nama lengkap atau kode unik yang mengidentifikasi arsip tersebut, memberikan kejelasan mengenai dokumen yang dimaksud.
- Jumlah berkas: Total jumlah berkas arsip yang telah kedaluwarsa, memudahkan pengguna dalam menilai beban kerja yang harus ditangani.

Dengan informasi yang komprehensif ini, pengguna akan lebih mudah membuat keputusan yang tepat terkait pengelolaan arsip, memastikan kepatuhan terhadap kebijakan internal serta regulasi yang berlaku. Selain itu, sistem notifikasi yang terstruktur ini juga berfungsi untuk meningkatkan efisiensi operasional dan mengurangi risiko kesalahan dalam pengelolaan dokumen.



Gambar 4. 43 Notifikasi Pemberitahuan Arsip Yang Habis Masa Retensinya

Gambar di atas menunjukkan contoh pesan notifikasi yang akan secara otomatis dikirimkan oleh *bot* Telegram setiap pagi pukul 07.00-08.00 WIB kepada pengguna yang telah mendaftar untuk menerima notifikasi. Pesan ini berisi informasi mengenai arsip yang telah melewati masa retensi, termasuk nama arsip dan jumlah berkas.. Dengan adanya informasi tambahan ini, pengguna dapat lebih memahami urgensi dari pengelolaan arsip yang tepat waktu.

Dengan sistem notifikasi yang terstruktur dan informatif ini, pengguna akan dapat lebih mudah mengambil keputusan yang tepat terkait pengelolaan arsip. Pada akhirnya, implementasi *bot* Telegram ini tidak hanya meningkatkan efisiensi operasional, tetapi juga mendukung kepatuhan terhadap regulasi yang berlaku, menjadikannya alat yang sangat berharga dalam manajemen arsip.

4.4 Kelebihan dan Kekurangan *Bot*

4.4.1 Kelebihan *Bot* :

- Tampilan *menu bot* terkesan sederhana sehingga mudah dipahami. Desain antarmuka yang intuitif memastikan bahwa pengguna, bahkan bagi pemula yang baru menggunakan *bot*, dapat dengan cepat memahami cara kerja fitur-fitur yang tersedia. Dengan navigasi yang jelas dan pemisahan yang rapi antara setiap kategori, pengguna tidak perlu mengalami kebingungan saat mencari informasi atau fungsi yang diinginkan.

Selain itu, panduan penggunaan yang tersedia di dalam *bot* membantu memberikan orientasi awal bagi para pengguna baru. Panduan ini mencakup petunjuk langkah demi langkah mengenai berbagai opsi yang ada, sehingga proses belajar menjadi lebih lancar dan efisien. Hal ini sangat penting, mengingat tidak semua pengguna memiliki latar belakang teknis yang mendalam, sehingga dengan adanya kemudahan ini, semua orang dapat merasa nyaman menggunakan *bot*.

- Tidak dibutuhkan biaya dalam pembuatan *bot*. Seluruh proses pembuatan *bot* Telegram tidak mengeluarkan biaya apapun, yang menjadikannya pilihan yang sangat ekonomis untuk berbagai kepentingan. Pengguna dapat memanfaatkan platform ini secara gratis, memungkinkan individu dan organisasi untuk mengimplementasikan solusi otomatisasi tanpa harus mengkhawatirkan anggaran yang terbatas.

Keberadaan *bot* Telegram yang mudah diakses dan tanpa biaya ini memberikan peluang luas bagi usaha kecil dan pemula untuk meningkatkan efisiensi operasional mereka.

Dengan memanfaatkan *bot*, mereka dapat mengotomatisasi tugas-tugas yang berulang, seperti pengiriman pesan pengingat atau informasi penting, sehingga dapat mengalokasikan waktu dan *sumber* daya yang berharga pada aspek lain.

- Privasi dan keamanan pengguna dijamin keamanannya oleh Telegram, yang telah mengembangkan berbagai fitur canggih untuk melindungi data penggunanya. Karena Telegram merupakan platform yang terkenal memperhatikan privasi para penggunanya, banyak orang merasa lebih aman saat berkomunikasi menggunakan aplikasi ini. Telegram menawarkan enkripsi end-to-end pada pesan yang dikirim melalui fitur "Secret Chats," sehingga hanya pengirim dan penerima yang dapat membaca isi pesan tersebut.
- Selain itu, Telegram juga menyediakan opsi untuk mengatur waktu kedaluwarsa pesan yang dikirim, yang semakin meningkatkan tingkat privasi. Dengan fitur ini, pengguna dapat mengontrol seberapa lama pesan tetap tersedia di perangkat penerima, yang tentunya membantu mengurangi risiko penyebaran informasi yang tidak diinginkan. Di samping itu, Telegram tidak menyimpan data pesan di server selamanya, sehingga informasi pengguna tetap terlindungi.
- Memberikan kemudahan bagi pengguna dalam mengontrol arsip inaktif yang sudah habis masa retensinya adalah salah satu fitur unggulan yang ditawarkan oleh platform ini. Karena *bot* akan memberikan notifikasi secara otomatis kepada pengguna terkait arsip yang sudah habis masa retensinya sesuai dengan waktu yang sudah ditentukan, pengguna dapat dengan mudah mengelola dan mengambil

keputusan yang tepat terkait arsip tersebut. Proses ini tidak hanya mengurangi risiko kehilangan informasi penting, tetapi juga mengoptimalkan penggunaan ruang penyimpanan yang tersedia.

Dengan adanya sistem pengingat yang berbasis otomatis ini, pengguna tidak lagi perlu mengingat tanggal ataupun melakukan pengecekan secara manual terhadap arsip-arsip yang ada. Dalam dunia yang bergerak cepat ini, efisiensi waktu menjadi sangat berharga, dan fitur notifikasi ini hadir untuk menangani kebutuhan tersebut dengan cara yang sangat efektif. Pengguna dapat menetapkan parameter tertentu, seperti frekuensi notifikasi dan jenis arsip yang ingin dipantau, sehingga pengalaman pengelolaan arsip menjadi lebih personalized.

- *Bot* dirancang untuk memiliki kemudahan dalam kustomisasi, yang menjadi salah satu keunggulannya. Pengguna akan menemukan bahwa proses penambahan fitur ke dalam *bot* cukup sederhana dan tidak memerlukan keterampilan teknis yang mendalam. Hal ini disebabkan oleh upaya Telegram yang secara aktif mendukung para penggunanya dengan berbagai fasilitas dan alat yang diperlukan untuk melakukan modifikasi. Dengan demikian, setiap pengguna memiliki kebebasan untuk menyesuaikan *bot* sesuai dengan kebutuhan serta preferensi spesifik mereka.
- Integrasi yang luas menjadi salah satu aspek yang sangat menonjol dari *bot* Telegram. Keunggulan ini memberikan kemudahan bagi pengembang untuk menghubungkan *bot* dengan berbagai platform dan layanan pihak ketiga yang

berbeda. Fleksibilitas *bot* Telegram memungkinkan interaksi yang tak terbatas, mulai dari aplikasi manajemen proyek hingga layanan pembayaran, sehingga pengguna dapat memanfaatkan banyak fitur tanpa harus meninggalkan platform Telegram.

- Kecepatan pemrosesan data. Pada metode pencarian arsip yang sudah habis masa retensinya yang masih menggunakan cara manual petugas menghabiskan waktu hampir 2-3 jam untuk menyelesaikan satu file daftar arsip inaktif. Sedangkan *bot* Telegram dapat menyelesaikan hal tersebut hanya dalam waktu 2-3 detik. Hal ini dapat memberikan efisiensi kerja sebesar 99,97%

4.4.2 Kekurangan *Bot* :

- 1) proses pembuatan *bot* dapat dianggap cukup rumit dan memakan waktu. Hal ini dimulai dari tahap persiapan yang memerlukan penyusunan *spreadSheet* yang teliti, di mana setiap detail harus diperhatikan dengan saksama agar potensi kesalahan dapat diminimalisir. Setelah tahapan tersebut, langkah selanjutnya adalah pembuatan *bot* pada platform Telegram, yang juga memerlukan pemahaman tentang berbagai bahasa pemrograman dan alat yang tersedia. Proses ini tidak hanya membutuhkan ketelitian, tetapi juga dedikasi dalam belajar untuk memahami cara kerja *bot* dan bagaimana mengintegrasikannya dengan sistem yang ada. Bagi sebagian orang, terutama yang tidak memiliki latar belakang teknis, tantangan ini bisa menjadi hambatan yang cukup besar.
- 2) Salah satu kekurangan *bot* adalah kebutuhan untuk terhubung dengan internet agar dapat menerima notifikasi dari *bot*. Hal ini disebabkan oleh fakta bahwa *bot* Telegram beroperasi dengan cara terhubung ke server, yang artinya setiap

pengiriman dan penerimaan pesan memerlukan akses jaringan yang stabil. Tanpa koneksi internet, pengguna tidak akan menerima pemberitahuan penting atau pesan yang mungkin diperlukan pada waktu yang kritis. Situasi ini dapat menjadi kendala, terutama di daerah dengan jaringan internet yang tidak stabil atau bagi pengguna yang sering bepergian ke lokasi dengan sinyal yang lemah. Selain itu, adanya ketergantungan pada koneksi internet juga memengaruhi pengalaman pengguna, karena potensi keterlambatan dalam menerima notifikasi bisa terjadi ketika jaringan mengalami gangguan.

- 3) Apabila petugas penerima pesan notifikasi dari Telegram sudah tidak bekerja lagi pada instansi yang bersangkutan, maka perlu dilakukan pengaturan ulang pada *bot* agar notifikasi tersebut dapat diterima oleh petugas yang masih aktif di instansi tersebut. Proses ini sering kali melibatkan beberapa langkah, termasuk identifikasi petugas baru yang akan menjadi penerima pesan dan memastikan bahwa informasi kontak mereka terintegrasi dengan sistem *bot* yang ada. Selain itu, perlu juga diperhatikan bahwa pengaturan ulang ini krusial untuk menjaga kesinambungan dalam komunikasi dan pengiriman informasi, sehingga tidak ada yang terlewatkan oleh staf yang bertanggung jawab. Jika pengaturan tidak dilakukan dengan cepat, risiko kehilangan notifikasi penting atau pesan mendesak dapat meningkat, yang pada akhirnya dapat berdampak negatif pada operasional instansi. Lebih jauh lagi, pengabaian terhadap proses pembaruan penerima notifikasi dapat menyebabkan ketidakpuasan di kalangan staf yang masih bekerja, terutama jika mereka tidak mendapatkan informasi yang diperlukan untuk menjalankan tugas mereka dengan efisien. Oleh karena itu, pengelolaan *bot* harus dilakukan secara proaktif, dengan

memperhatikan setiap perubahan dalam struktur organisasi dan memperbarui data penerima pesan secara berkala.