

Perancangan Aplikasi untuk Manajemen Rekaman IP CCTV pada FTP Server FreeBSD

Jonathan Chandra
Universitas Kristen Maranatha
joe.chd.pub@gmail.com

Abstrak - Penggunaan IP CCTV di masyarakat merupakan sebuah hal yang umum pada akhir-akhir ini. Penggunaannya pun beragam mulai dari pemantauan lingkungan dari tindak kejahatan hingga pemantauan sebagai tindakan preventif dalam kebencanaan. Jenis dan bentuk IP CCTV semakin beragam dan sebagian besar sudah memiliki kemampuan untuk terhubung ke jaringan komputer, salah satunya adalah IP CCTV murah yang memanfaatkan koneksi FTP dalam operasinya. FreeBSD merupakan salah satu sistem operasi yang digunakan pada infrastruktur jaringan komputer dan juga memiliki kemampuan untuk dapat digunakan pada sistem tertanam khususnya dalam ekosistem IoT. Salah satu kekurangan dari penggunaan FTP server umumnya adalah tidak adanya fitur manajemen berkas rekaman IP CCTV. Dengan adanya sebuah aplikasi yang mampu mengelola rekaman pada penyimpanan FTP server tentunya mempermudah dalam hal pengolahan dan terutama dalam efisiensi penggunaan tempat penyimpanan yang kapasitasnya terbatas. Hal ini terbukti berhasil dengan hadirnya aplikasi tersebut sehingga penyimpanan rekaman IP CCTV pada suatu FTP server dapat dikelola dengan baik.

Kata Kunci : ipcctv, ftp, manajemen rekaman, freebsd

Abstract - The usages of IP CCTV in community are becoming a common thing nowadays. The purpose of installing IP CCTV is vary widely from user to user, starting from crime activity monitoring up to disaster prevention. IP CCTV devices form and features are so diverse to choose from, one of them is a low cost IP CCTV camera that has capability to use FTP connections. FreeBSD is one of operating system found in computer network infrastructure and has the capability to be installed inside an embedded system especially when used on IoT ecosystem. One of weakness from using a common FTP server is lack of file management, especially for IP CCTV purpose. With an application that can manage IP CCTV recordings on an FTP storage can help in managing recordings easily and helps in dealing with limited storage efficiently. This is proven by that application a IP CCTV repository inside a FTP server can be managed well.

Keywords: ipcctv, ftp, recordings management, freebsd

I. PENDAHULUAN

Penggunaan CCTV di masyarakat pada saat ini merupakan hal sangat yang umum dan semakin marak belakangan ini. Hal ini berkaitan dengan banyaknya tindak kejahatan yang dapat terekam dalam bentuk rekaman digital (Ashby, 2017) sehingga dapat membantu pihak berwenang dalam hal melakukan investigasi lebih lanjut terkait barang bukti berupa rekaman digital (Ariananda, 2017). Selain penggunaannya dalam hal pemantauan keamanan lingkungan, penggunaan CCTV juga dapat digunakan sebagai alat bantu dalam hal pemantauan keadaan lingkungan. Contohnya seperti penggunaannya dalam memantau ketinggian air agar sehingga dapat dilakukannya tindakan preventif sehingga terhindar bencana banjir (Antari dkk., 2017). Keunggulan utama dari penggunaan IP CCTV adalah kemudahan dalam hal penyimpanan rekaman dalam format digital yang dapat disimpan pada perangkat komputer khususnya dalam ekosistem Internet of Things (Yan dkk., 2018). Dan seiring berkembangnya tren teknologi kearah *Internet of Things* maka penggunaan aplikasi kamera pemantau dengan

konektivitas seperti FTP (Harum dkk., 2018) akan semakin mudah dijumpai dan diaplikasikan seperti contohnya pada perangkat *wireless router* (Kao dkk., 2013).

Sistem operasi FreeBSD merupakan salah satu sistem operasi *open source* alternatif selain GNU Linux yang ditemukan dalam infrastruktur jaringan komputer. Sistem operasi ini juga memiliki keunggulan untuk dapat dipasangkan pada berbagai perangkat sistem tertanam seperti *single board computer*. Salah satu *single board computer* yang didukungnya yaitu Raspberry Pi (*FreeBSD/ARM Project*, 2021). Sistem operasi FreeBSD pada penelitian ini bertugas sebagai *host* aplikasi FTP server yang bertujuan untuk menyimpan rekaman dari IP CCTV.

Aplikasi untuk memantau rekaman CCTV telah banyak dibuat dan dilakukan pengembangan pada berbagai jenis sistem operasi dengan tujuannya yang beragam baik itu gratis maupun berbayar.

Jenis pertama seperti pengembangan aplikasi pemantauan CCTV yang dilakukan pada sistem operasi Microsoft Windows. Pengembangan aplikasi yang dilakukan oleh (Hadiwijaya dkk.,

2014) adalah merancang sebuah aplikasi pemantauan CCTV dengan menggunakan Visual Studio untuk memantau keadaan ruangan. Contoh lainnya masih pada jenis sistem operasi Microsoft Windows adalah pengembangan aplikasi pemantauan CCTV yang dilakukan oleh (Robiyanto, 2019) dengan memanfaatkan teknologi server FTP (FileZilla) sama seperti yang dilakukan pada penelitian ini namun dengan perbedaan di tujuan akhirnya berupa notifikasi kepada pengguna aplikasi.

Aplikasi pemantauan IP CCTV pada sistem operasi lainnya seperti di GNU Linux cukup banyak ragam dan jenisnya. Beberapa di antaranya cukup populer digunakan salah satunya seperti aplikasi ZoneMinder yang digunakan oleh (Saputra dkk., 2013) dalam membuat sebuah sistem pengawasan ruangan. Aplikasi ZoneMinder sebetulnya bisa dipasang pada sistem operasi FreeBSD namun memiliki beberapa kelemahan jika dibandingkan dengan aplikasi yang dirancang pada penelitian ini. Pertama kebutuhan perangkat keras yang cukup tinggi seperti yang diutarakan oleh (Wijayanto & Irsyadi, 2015) pada penelitiannya, sehingga pengaplikasian pada perangkat keras terbatas seperti Raspberry Pi akan menjadi tantangan tersendiri terutama dengan jenis arsitektur prosesor yang jelas berbeda. Kedua adalah masalah fungsionalitas aplikasi itu sendiri jika tetap menggunakan FTP server dari sistem operasi FreeBSD karena aplikasi ZoneMinder belum mendukung pemantauan direktori FTP secara khusus.

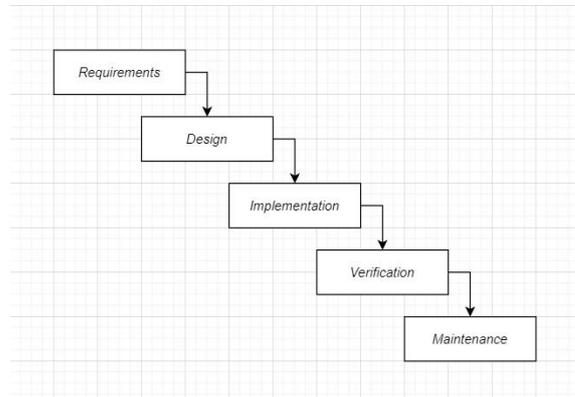
Maka dengan pilihan aplikasi yang terbatas untuk sistem operasi khususnya FreeBSD dan agar memenuhi tujuan untuk dapat melakukan pemantauan tanpa memerlukan perangkat keras dengan spesifikasi yang tinggi maka pilihan pengembangan sebuah aplikasi yang memanfaatkan FTP server adalah salah satu solusinya.

Salah satu kekurangan dari aplikasi FTP server umumnya adalah kurangnya fitur manajemen berkas rekaman IP CCTV yang ada pada sisi server. Dengan dibuatnya sebuah aplikasi untuk hal ini tentunya akan memudahkan pengguna untuk melihat rekaman pada waktu-waktu tertentu yang diinginkan. Selain itu aplikasi ini juga akan memudahkan dalam hal pembersihan rekaman yang sudah lama dan tidak relevan lagi sehingga terjadi efisiensi dalam penggunaan tempat penyimpanan yang terbatas.

II. METODOLOGI PENELITIAN

Metode pengembangan perangkat lunak ini menggunakan metode *waterfall*, hal ini didasari oleh sifat pengembangan perangkat lunak yang berurutan, proses tahap per tahap dengan tenggat waktu yang jelas. Sehingga memudahkan

pengembang dalam melakukan dokumentasi pengembangan perangkat lunak per bagian (A. A. Adenowo & Adenowo, 2013) dan juga memudahkan dalam hal pengembangan aplikasi perangkat lunak secara mandiri.



Sumber: Penelitian (2020).

Gambar 1. Metode *waterfall* yang digunakan dalam pengembangan.

Untuk pengujian perangkat lunak digunakan metode *black box testing*. Metode *black box testing* adalah teknik pengujian perangkat lunak dengan mengedepankan aspek pengujian pada sisi fungsionalitas dari perangkat lunak itu sendiri. Teknik pengujian ini menguji tanpa membahas lebih lanjut mengenai isi maupun cara kerja dari perangkat lunak tersebut secara spesifik (Verma dkk., 2017). Metode pengujian pada penelitian ini didasari oleh pengecekan fungsionalitas perangkat lunak berdasarkan dengan perumusan masalah dan kebutuhan sistem yang sudah didefinisikan terlebih dahulu.

1. Analisis Kebutuhan Sistem

Secara garis besar kebutuhan sistem didasari oleh beberapa poin, sesuai dengan pembahasan pada bagian pendahuluan. Berikut ini adalah penjabaran mengenai kebutuhan sistem yang sudah dianalisa dan dipetakan.

Pertama adalah kebutuhan fungsi dari aplikasi berupa kemampuan untuk membaca pola dari nama berkas *video* IP CCTV.

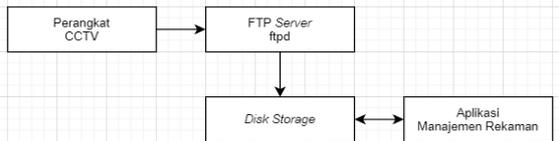
Kedua adalah kebutuhan fungsi dari aplikasi berupa kemampuan untuk membuka dan menayangkan rekaman *video* IP CCTV yang tersimpan pada *disk storage*.

Ketiga adalah kemampuan yang dimiliki oleh aplikasi untuk dapat membaca waktu dan penanggalan dari berkas *video* IP CCTV untuk dapat diproses lebih lanjut agar dapat ditentukan operasinya. Nantinya aplikasi memutuskan apakah suatu berkas *video* yang disimpan telah melebihi waktu yang ditentukan atau belum. Jangka waktu penyimpanan berkas ini tentunya diatur pada aplikasi.

Keempat adalah kemampuan aplikasi untuk melakukan operasi penghapusan berkas *video* IP CCTV pada *disk storage*.

2. Blok Diagram

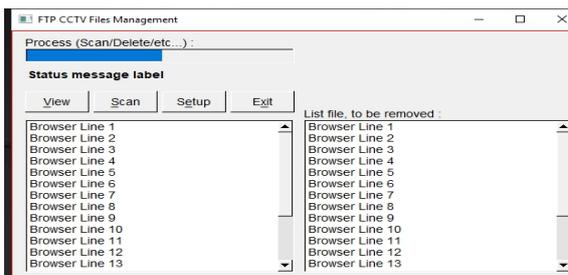
Pada gambar berikut ini akan menampilkan cara kerja sistem mulai dari perangkat IP CCTV yang sudah terhubung melalui infrastruktur jaringan komputer, dalam hal ini contohnya adalah perangkat *router* WiFi, lalu menuju ke komputer *server* yang sudah terpasang aplikasi *FTP server*. *FTP server* akan menyimpan data berupa rekaman *video* pada *disk storage*, yang nantinya akan diolah dengan aplikasi manajemen rekaman yang dikembangkan pada penelitian ini.



Sumber: Penelitian (2020).
Gambar 2. Blok diagram.

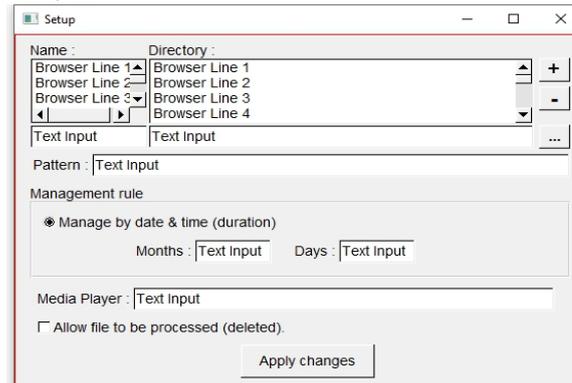
3. Desain Antarmuka

Desain antarmuka dari aplikasi ini terbagi menjadi 3 *form* dimana masing-masing *form* memiliki fungsi dan tujuannya masing-masing. Pembagian tampilan ini bertujuan agar fungsionalitas dari aplikasi lebih terorganisir dan juga terstruktur dalam pemrogramannya. Tampilan *form* pertama adalah *form* utama. Dalam tampilan *form* utama ini terbagi kembali menjadi 2 bagian. Bagian pertama merupakan tampilan yang memuat *menu* navigasi dalam aplikasi yang diwakili oleh 2 buah tombol yang bertuliskan View dan Setup. Dari masing-masing tombol tersebut akan membuka *form* masing-masing sesuai dengan navigasinya. Bagian kedua adalah fungsi *scan* yang merupakan inti utama dari fungsionalitas aplikasi yang diwakili oleh tombol Scan. Fungsi *scan* tersebut akan memberikan informasi tambahan yang dapat dilihat pada kedua buah *list box*. *List box* sebelah kiri menampilkan keseluruhan berkas yang terdeteksi oleh aplikasi sebagai berkas *video* IP CCTV. Pada *list box* sebelah kanan menampilkan berkas mana saja yang diproses (dalam hal ini adalah proses penghapusan). Selain informasi berupa daftar berkas, fungsi *scan* ini juga menampilkan kemajuan dalam pemrosesannya dalam komponen *progress bar* dan juga komponen *label status*.



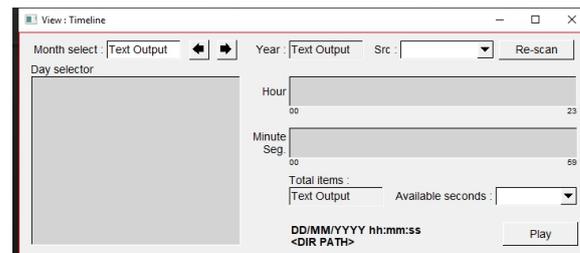
Sumber: Penelitian (2020).
Gambar 3. Desain tampilan *form* utama.

Form berikutnya adalah *form setup* yang memiliki tugas sebagai tempat pengaturan aplikasi dalam hal melakukan fungsinya. Pengaturan yang dapat diatur pada *form setup* ini mulai dari jumlah kamera IP CCTV, pola penamaan berkas yang akan dideteksi oleh aplikasi, durasi penyimpanan berkas (dalam hari dan juga bulan) hingga pemilihan aplikasi pemutar *video* dan opsi penghapusan berkas secara permanen. Untuk mengakses *form* ini dapat melalui tombol navigasi Setup pada *form* utama. Pada gambar berikut ini merupakan desain tampilan dari *form setup* tersebut.



Sumber: Penelitian (2020).
Gambar 4. Desain tampilan *form setup*.

Form selanjutnya adalah *form view* yang bertugas sebagai antarmuka dalam hal pemilihan dan juga pemutaran berkas *video* yang tersimpan pada *FTP server* berdasarkan tanggal dan waktu. Desain tampilan antarmuka *form view* berisikan beberapa komponen dapat dilihat pada gambar berikut.



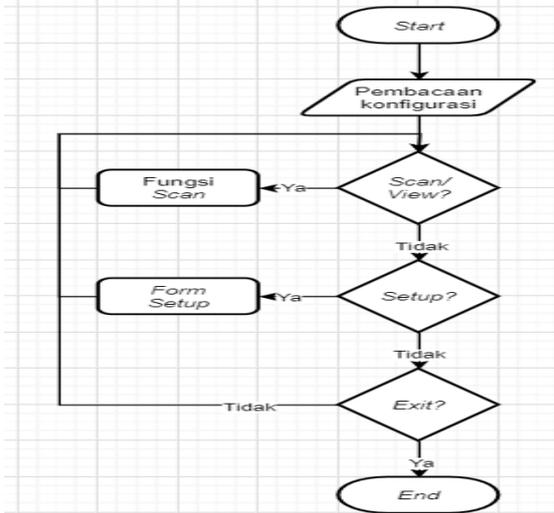
Sumber: Penelitian (2020).
Gambar 5. Desain tampilan *form view*.

Pertama komponen sejenis kalender yang dapat melakukan navigasi untuk melihat rekaman per bulannya. Di atasnya terdapat seleksi pemilihan bulan rekaman yang sudah tersimpan. Pada sisi kanan atas adalah pemilihan rekaman perangkat CCTV, jika yang digunakan pada sistem lebih dari 1 buah CCTV. Pada bagian kanan tengah hingga bawah adalah detail rekaman untuk satu hari yang dipilih pada *widget* kalender sebelumnya, terbagi menjadi jam dan detik. Pada bagian tersebut, pengguna dapat memilih waktu rekaman yang ingin diputar. Tombol Play pada sisi kanan bawah akan membuka aplikasi

pemutar *video* untuk melihat rekaman yang terpilih.

4. Flowchart Sistem

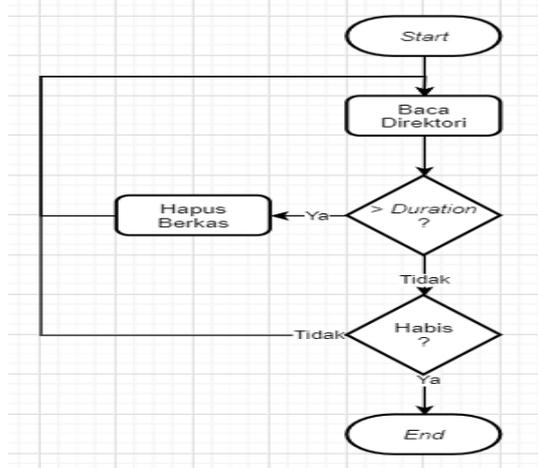
Pada bagian ini akan dijelaskan mengenai cara kerja perangkat lunak secara bertahap dengan menggunakan dua buah *flowchart*. *Flowchart* yang ditampilkan pada gambar 6 dan 7 akan menjelaskan mengenai gambaran besar dari cara kerja sistem dan juga fungsi inti dari aplikasi ini yaitu fungsi *scan*.



Sumber: Penelitian (2020).

Gambar 6. *Flowchart* sistem kerja perangkat lunak.

Cara kerja aplikasi secara garis besar dapat terlihat pada gambar 6 *flowchart* sistem kerja perangkat lunak. Aplikasi awalnya melakukan pembacaan konfigurasi yang berisikan nama-nama IP CCTV yang terdaftar beserta direktorinya masing-masing lalu diikuti dengan pembacaan aturan pola untuk melakukan *search*, pembacaan alamat aplikasi eksternal *media player* hingga pembacaan opsi pengolahan berkas. Setelah itu aplikasi melakukan perintah sesuai dengan arahan dari pengguna, apakah akan melakukan *scan* berkas, melihat rekaman atau bahkan membuka pengaturan aplikasi kembali.



Sumber: Penelitian (2020).

Gambar 7. *Flowchart* fungsi *scan*.

Flowchart fungsi *scan* menjelaskan mengenai tata cara aplikasi dalam melakukan operasi *scan* dan juga bagaimana tahap penghapusan berkas itu terjadi. Pertama-tama aplikasi akan membuka direktori tempat penyimpanan berkas rekaman *video* CCTV satu per satu lalu dilanjutkan pemrosesan berkas per berkas dengan iterasi *loop*. Setiap iterasi, akan terjadi proses pengecekan terhadap durasi penyimpanan untuk masing-masing berkas. Jika durasi suatu berkas melebihi dari aturan penyimpanan pada aplikasi, maka akan terjadi pengolahan lebih lanjut yakni penghapusan berkas. Hal ini berlangsung secara terus menerus hingga jumlah berkas dalam direktori penyimpanan *video* rekaman selesai diperiksa secara keseluruhan.

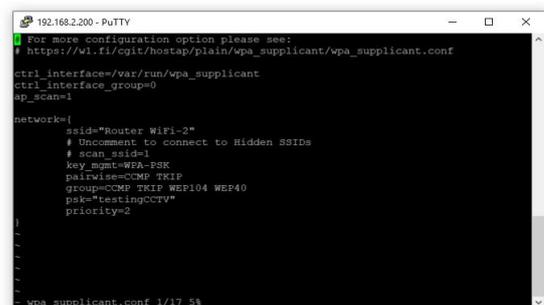
III. HASIL DAN PEMBAHASAN

Pengujian perangkat lunak akan terdiri dari dua bentuk pengujian. Pengujian pertama akan membahas mengenai tampilan aplikasi dalam hal penggunaannya berupa navigasi dalam aplikasi, penggunaan tombol dan sebagainya. Pengujian kedua adalah pengujian dengan metode *black box testing* yang bertujuan untuk menguji fungsionalitas atau cara kerja dari aplikasi tersebut berdasarkan hasil yang dikehendaki, apakah sudah sesuai atau tidak.

1. Pengaturan IP CCTV

Sebelum dilakukannya pengujian, perangkat IP CCTV yang akan digunakan perlu diatur terlebih dahulu agar dapat menggunakan koneksi FTP yang sudah disiapkan sebelumnya. Terdapat 2 hal yang harus dilakukan pengaturan terhadap perangkat IP CCTV. Pertama adalah konektivitas jaringan, pada penelitian ini jenis jaringan yang digunakan merupakan jaringan nirkabel. Kedua yang perlu dilakukan adalah pengaturan identitas akses FTP pada perangkat IP CCTV sehingga perangkat dapat melakukan akses terhadap *server* FTP di sistem operasi FreeBSD yang sudah disediakan.

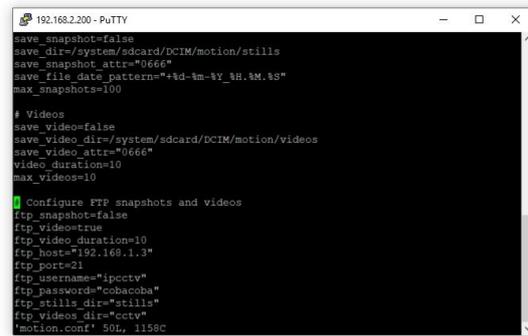
Pada gambar berikut merupakan cuplikan pengaturan pertama yaitu jaringan nirkabel yang dilakukan pada perangkat IP CCTV pada penelitian ini.



Sumber: Penelitian (2020).

Gambar 8. Pengaturan jaringan nirkabel pada perangkat IP CCTV.

Gambar 9 merupakan cuplikan pengaturan kedua yaitu pengaturan identitas akses server FTP terhadap perangkat IP CCTV.



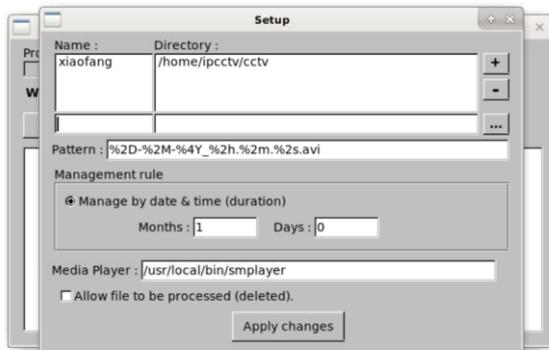
Sumber: Penelitian (2020).

Gambar 9. Pengaturan identitas akses FTP pada perangkat IP CCTV.

Setelah selesainya dilakukan pengaturan kedua hal tersebut, maka perangkat IP CCTV sudah dapat digunakan secara langsung. Setiap terjadi kejadian yang tergolong *motion detected movement*, maka perangkat IP CCTV akan mengirimkan sebuah rekaman berbentuk berkas dengan format *video* AVI dengan durasi yang sudah ditentukan, sesuai dengan pengaturan pada tahap kedua.

2. Pengujian Fungsionalitas Antarmuka

Berikut ini merupakan pengujian terhadap tampilan aplikasi yang telah dibuat. Pembahasan tampilan akan dimulai dari *form* utama, *form setup* hingga *form view*.



Sumber: Penelitian (2020).

Gambar 10. Pengujian tampilan *form* utama.

Gambar 10 di atas menunjukkan tampilan aplikasi ketika di *running* dengan tampilan utamanya berupa *form* utama. Pada gambar tersebut terlihat operasi atau fungsi *scan* telah selesai dilakukan dan menunjukkan kondisi berkas mana saja yang terdeteksi dan juga berkas yang terproses pada *list box* kiri dan kanan.

Progress bar juga telah berhasil menunjukkan tingkat kemajuan proses deteksi *scan* dan proses berkas namun hal ini tidak dapat terlihat secara langsung pada gambar tersebut karena gambar diambil setelah fungsi *scan* selesai.

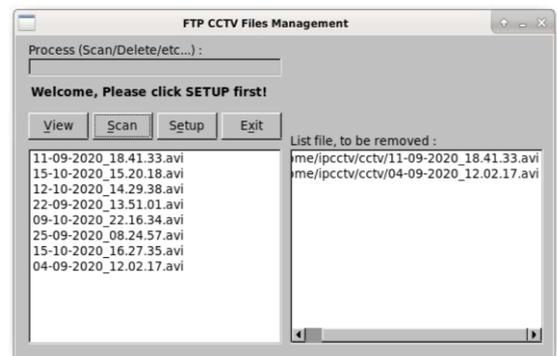
Selain menguji fungsi tombol *Scan* dan juga tombol *Exit*, pengujian navigasi aplikasi dari tombol *View* dan *Setup* akan ditunjukkan pada tampilan *form* di kedua gambar selanjutnya.



Sumber: Penelitian (2020).

Gambar 11. Pengujian tampilan *form setup*.

Gambar 11 menunjukkan tampilan *form setup* setelah dibuka oleh *form* utama. Terlihat pengaturan yang dilakukan berupa penambahan satu sumber IP CCTV beserta direktori penyimpanannya. Selanjutnya adalah pengaturan yang digunakan dalam rangka pendeteksian pola berkas *video*. Lalu berikutnya adalah pengaturan durasi penyimpanan yang diatur selama 1 bulan lamanya. Pengaturan lainnya berupa aplikasi *media player* yang digunakan dan juga opsi penghapusan secara permanen.



Sumber: Penelitian (2020).

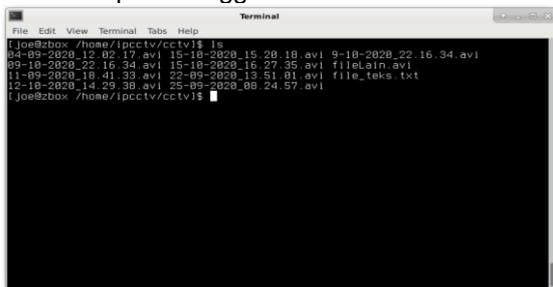
Gambar 12. Pengujian tampilan *form view*.

Gambar 12 menunjukkan tampilan *form view* setelah dibuka dari *form* utama. Tampilan tersebut berhasil menunjukkan berkas *video* yang ada pada tempat penyimpanan IP CCTV ini. Sisi kiri *form* menunjukkan hari rekaman video yang dapat dipilih (berwarna oranye) dan juga hari yang terpilih (berwarna hijau). Pada sisi kanannya pun demikian cara kerjanya hanya seleksinya menjadi jam, menit hingga detik yang dapat dipilih. Pemilihan untuk bulan dan sumber IP CCTV pun dapat dilakukan sesuai dengan rancangan.

3. Fungsi Utama dan Decision Table Testing

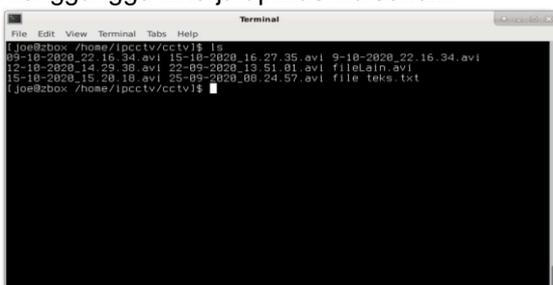
Setelah pengujian tampilan aplikasi manajemen IP CCTV, selanjutnya adalah pengujian fitur-fitur aplikasi ini menggunakan metode *black box testing*. Pada metode *black box testing*, digunakan teknik *decision table testing* yang merupakan pengujian berdasarkan perbandingan *input* dan aksi yang dilakukan terhadap *input* tersebut. Metode ini akan dibahas pada tabel 1 dan 2 setelah pembahasan hasil aplikasi jika dilihat dalam segi penyimpanan berkas.

Berikut ini adalah dua buah tampilan kondisi dari tempat penyimpanan berkas jika dilihat secara fungsionalitas aplikasi dalam mengolah berkas. Gambar yang diambil pada pengujian berikut dilakukan pada tanggal 15 Oktober 2020.



Sumber: Penelitian (2020).

Gambar 13. Kondisi direktori penyimpanan sebelum dilakukannya *scan* dan *processing*. Pada gambar tersebut terlihat pada penyimpanan berkas atau direktori cctv terdapat banyak berkas *video* rekaman yang sudah diunggah oleh perangkat IP CCTV melalui aplikasi *FTP server*. Berkas-berkas *video* rekaman tersebut hanya ada berdasarkan *motion detection* yang sudah diatur sedemikian rupa sebelumnya. Selain itu dalam pengujian ini ditambahkan juga beberapa berkas yang tidak memenuhi kriteria pencarian berkas oleh aplikasi. Hal ini terlihat dengan adanya beberapa berkas seperti: "9-10-2020_22.16.34.avi", "fileLain.avi", dan "file_teks.txt". Tujuan dengan penambahan beberapa berkas tersebut untuk menguji fungsionalitas aplikasi, apakah adanya faktor lain seperti berkas yang tidak relevan mengganggu kinerja aplikasi itu sendiri.



Sumber: Penelitian (2020).

Gambar 14. Kondisi direktori penyimpanan setelah dilakukannya *scan* dan *processing*.

Setelah melalui tahap *scan* di aplikasi manajemen rekaman, maka terlihat kondisi penyimpanan berkas sudah berubah (gambar 14). Terlihat pada gambar tersebut, berkas rekaman pada bulan september sebelum tanggal 15, terhapus. Sedangkan setelah tanggal 15 rekaman yang tersisa tetap utuh tidak tersentuh. Lalu berkas lain yang tidak relevan dengan aturan pembacaan berkas di aplikasi, tetap ada dan tidak disentuh sama sekali.

Selesai melihat perubahan pada media penyimpanan berkas *video* rekaman, selanjutnya adalah pemetaan hasil pengujian ke dalam *decision table testing* berikut ini. Tabel yang digunakan pada pengujian ini terdapat 2 bagian. Masing-masing tabel memiliki peranannya masing-masing. Tujuan dari penggunaan 2 tabel untuk *decision table testing* ini agar dapat menambahkan keterangan per inputnya dan juga pemetaan yang lebih mudah dan rapi.

Tabel 1. Kumpulan macam-macam pola berkas beserta keterangannya masing-masing.

No	Nama Berkas	Keterangan
1	04-09-2020_12.02.17.avi	Contoh pertama
2	11-09-2020_18.41.33.avi	Contoh kedua
3	22-09-2020_13.51.01.avi	Contoh ketiga
4	25-09-2020_08.24.57.avi	Contoh keempat
5	09-10-2020_22.16.34.avi	Contoh kelima
6	12-10-2020_14.29.38.avi	Contoh keenam
7	15-10-2020_15.20.18.avi	Contoh ketujuh
8	15-10-2020_16.27.35.avi	Contoh kedelapan
9	9-10-2020_22.16.34.avi	Contoh tidak sesuai aturan pola
10	fileLain.avi	Berkas dengan format sama namun beda pola
11	file_teks.txt	Berkas dengan jenis berbeda sama sekali

Sumber: Penelitian (2020).

Pada tabel 1 terlihat banyaknya jumlah berkas yang ada pada media penyimpanan rekaman *video* IP CCTV, yaitu 11 berkas. Sesuai dengan yang sudah dijelaskan sebelumnya, terdapat beberapa berkas yang ditambahkan dan tidak relevan dengan proses pencarian berkas pada aplikasi. Hal ini dapat dilihat mulai dari berkas nomor 9 hingga 11. Sedangkan berkas yang memiliki kriteria lengkap sesuai dengan aturan pencarian berkas di aplikasi mulai dari berkas nomor 1 hingga nomor 8.

Tabel 2. Decision table testing.

	Pola Berkas										
	1	2	3	4	5	6	7	8	9	10	11
Kondisi:											
Terdeteksi	v	v	v	v	v	v	v	v			
Tidak terdeteksi									v	v	v
Aksi:											
Hapus	v	v									
Lewati			v	v	v	v	v	v			

Sumber: Penelitian (2020).

Tabel kedua ini menjelaskan hal utama dari *decision table testing*. Terlihat pada tabel adanya perbandingan antara *input* berupa pola berkas yang sudah dijabarkan pada tabel pertama dan *decision* yang dilakukan oleh aplikasi.

Pada pengujian ini terlihat bahwa seluruh *input* dengan pola berkas yang benar dapat diolah dengan baik oleh aplikasi. Terlihat mulai dari pola berkas 1 hingga 8 masuk ke dalam kondisi terdeteksi atau cocok, sedangkan sisanya masuk ke dalam kondisi tidak terdeteksi atau tidak cocok.

Dari delapan kondisi tersebut dilakukan pengolahan lebih lanjut oleh aplikasi manajemen rekaman dan pada akhirnya setelah melakukan proses perbandingan, terdapat 2 *input* pola yang mengalami operasi penghapusan karena melebihi aturan durasi waktu penyimpanan berkas (1 bulan lamanya). Sedangkan berkas rekaman sisanya tidak mengalami operasi apapun, sehingga termasuk ke dalam aksi lewati oleh aplikasi.

Dari hasil tampilan kondisi media penyimpanan berkas rekaman *video* CCTV dan juga berdasarkan informasi dari kedua tabel *decision table testing*, dapat disimpulkan bahwa pengujian aplikasi manajemen rekaman CCTV ini telah berhasil dan dapat memenuhi kriteria kebutuhan sistem pada penelitian ini.

IV. KESIMPULAN

Tahap pengembangan aplikasi mulai dari penjabaran kriteria kebutuhan sistem lalu dilanjutkan dengan melakukan desain aplikasi beserta cara kerjanya beserta tahap pengujian yang sudah dilalui, maka dapat disimpulkan bahwa perancangan aplikasi manajemen rekaman IP CCTV dengan aplikasi FTP *server* di sistem operasi FreeBSD dapat dilakukan dengan baik.

Hal ini terbukti dengan hasil dari metode pengujian *black box testing* yang menguji fungsionalitas cara kerja dari aplikasi yang mampu mengelola kumpulan berkas-berkas rekaman *video* IP CCTV. Terlihat dengan jelas bahwa berkas yang memiliki nama dengan aturan kombinasi penamaan sesuai dengan kriteria arahan aplikasi, dapat terolah dengan baik.

Dengan hasil penelitian ini yang berupa aplikasi pemantauan IP CCTV pada sistem operasi FreeBSD khususnya yang memanfaatkan penyimpanan FTP *server* tentunya akan membuka kesempatan pengembangan baru selain yang sudah dilakukan oleh penelitian lainnya contohnya pada sistem operasi umum seperti halnya pada Microsoft Windows.

Pada pengembangan penelitian selanjutnya proses pemilihan rekaman tidak hanya berdasarkan waktu dan tanggal saja, namun berkas *video* dapat diolah dengan kecerdasan

buatan sehingga penyeleksian akan jauh lebih efisien dan memaksimalkan penggunaan tempat penyimpanan.

V. REFERENSI

- Adenowo, A. A. A., & Adenowo, B. A. (2013). *Software Engineering Methodologies: A Review of the Waterfall Model and Object-Oriented Approach*. 4(7), 8.
- Antari, T., Tentua, M. N., Si, S., & Sari, M. W. (2017). RANCANG BANGUN SISTEM PEMANTAU KETINGGIAN AIR SUNGAI MENGGUNAKAN MIKROKONTROLLER DAN CCTV BERBASIS WEB (Studi Kasus di Pusat Pengendalian Operasi Badan Penanggulangan. 5.
- Ariananda, R. A. (2017). Penggunaan Alat Bukti Rekaman CCTV (Closed Circuit Television) dalam Proses Peradilan Pidana. 69.
- Ashby, M. P. J. (2017). The Value of CCTV Surveillance Cameras as an Investigative Tool: An Empirical Analysis. *European Journal on Criminal Policy and Research*, 23(3), 441–459. <https://doi.org/10.1007/s10610-017-9341-6>
- FreeBSD/ARM Project. (2021). <https://www.freebsd.org/platforms/arm.html>
- Hadiwijaya, B., Zahra, A. A., & Darjat. (2014). PERANCANGAN APLIKASI CCTV SEBAGAI PEMANTAU RUANGAN MENGGUNAKAN IP CAMERA. *TRANSIENT*, 3(2), 6.
- Harum, N. binti, Faeq, M., Azma, N., & Anawar, S. (2018). Smart Surveillance System using Background Subtraction Technique in IoT Application. *International Journal of Advanced Computer Science and Applications*, 9(12). <https://doi.org/10.14569/IJACSA.2018.091218>
- Kao, T., Lee, Y., Lin, Y., Wu, K., & Hwang, L. (2013). Design of video surveillance in wireless router. *2013 International Symposium on Next-Generation Electronics*, 287–290. <https://doi.org/10.1109/ISNE.2013.6512346>
- Robiyanto, R. (2019). APLIKASI KEAMANAN RUANGAN MENGGUNAKAN TEKNOLOGI CCTV DAN SMS GATEWAY BERBASIS DELPHI. *Jurnal Ilmiah Indonesia*, 4(4), 9.
- Saputra, A. S., Christyono, Y., & Isnanto, R. R. (2013). PERANCANGAN SISTEM PENGAWASAN RUANGAN DENGAN KAMERA IP MENGGUNAKAN

- SISTEM OPERASI LINUX.
TRANSIENT, 2(3), 6.
- Verma, A., Department of Computer Science, Amity University, Gurgaon, India, Khatana, A., Department of Computer Science, Amity University, Gurgaon, India, Chaudhary, S., & Department of Computer Science, Amity University, Gurgaon, India. (2017). A Comparative Study of Black Box Testing and White Box Testing. *International Journal of Computer Sciences and Engineering*, 5(12), 301–304.
<https://doi.org/10.26438/ijcse/v5i12.301304>
- Wijayanto, A., & Irsyadi, F. Y. (2015). *PENGEMBANGAN APLIKASI SISTEM MONITORING KEAMANAN BERBASIS LINUX, MENGGUNAKAN CCTV DAN SMS GATEWAY*. UNIVERSITAS MUHAMMADIYAH SURAKARTA.
- Yan, H., Li, X., Wang, Y., & Jia, C. (2018). Centralized Duplicate Removal Video Storage System with Privacy Preservation in IoT. *Sensors*, 18(6), 1814.
<https://doi.org/10.3390/s18061814>