**Sistem Informasi Controlling Sertifikasi Awak Pesawat**

**Studi Kasus Pada PT.Kalstar Aviation**

# HALAMAN JUDUL

**SKRIPSI**

Ditulis untuk memenuhi sebagian persyaratan dalam

menempuh gelar sarjana Komputer

Jenjang Pendidikan Strata – 1

****

**Oleh:**

**SAEFUL MAHMUD**

**NIM:13.3.0132**

**JURUSAN TEKNIK INFORMATIKA**

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER**

**MUHAMMADIYAH JAKARTA**

**2017**

# 

# HALAMAN PERSETUJUAN

**SKRIPSI**

**Sistem Informasi Controlling Sertifikasi Awak Pesawat**

**Studi Kasus Pada PT.Kalstar Aviation**

**SAEFUL MAHMUD**

**NIM: 13.3.0132**

**Jakarta, 22 November 2017**

|  |  |
| --- | --- |
| **Menyetujui** | **Menyetujui** |
| **Hendarto, M.Kom**  **Ka. Prodi TI** | **M.Arief Sutisna, S.Kom**  **Dosen Pembimbing** |

**Mengetahui**

**Wakil Ketua Bidang Akademik**

**STMIK Muhammadiyah Jakarta**

|  |
| --- |
| **M.Zen Fitriansyah, S,Kom**  **NBM:841537** |

# HALAMAN PENGESAHAN

**SKRIPSI**

**Sistem Informasi Controlling Sertifikasi Awak Pesawat**

**Studi Kasus Pada PT.Kalstar Aviation**

**SAEFUL MAHMUD**

**NIM: 13.3.0132**

**Dipertahakan di depan penguji**

**Tanggal, 22 November 2017**

|  |  |  |
| --- | --- | --- |
| **SUSUNAN PENGUJI** | **TANDA TANGAN** | **TANGGAL** |
| Hendarto, M.Kom  (Penguji I) | ................................ | ............................. |
| Habibah Nurfauziah, S.Kom,M.Si  (Penguji II) | ................................ | ............................. |
| Nur Rachma, M.Kom  (Penguji III) | ................................ | ............................. |

**Jakarta, 22 November 2017**

**Mengetahui**

**Ketua STMIK Muhammadiyah Jakarta**

**Faiz Rafdhi CH, M.Kom**

**NBM:734048**

# HALAMAN PERNYATAAN

Yang bertanda tangan dibawah ini:

Nama : SAEFUL MAHMUD

NIM : 13.3.0132

Program Pendidikan : Strata Satu(S1)

Program Studi : Teknik Informatika

Dengan Ini Menyatakan Bahwa:

1. Skripsi ini merupakan hasil karya penulis sendiri dan hasil plagiasi. Jika dikemudian hari diketahui ternyata hasil karya orang lain, Saya bersedia menerima sanksi sesuai Undang-undang nomor 20 tahun 2003 pasal 25 tentang Sistem Pendidikan Nasional yaitu berupa pencabutan gelar sarjana dan ijazah.
2. Memberikan ijin kepada pihak **STMIK Muhammadiyah Jakarta Hak Bebas Royalti Non-Ekskutif**  atas karya ilmiah yang berjudul ”*Sistem Informasi Controlling Sertifikasi Awak Pesawat – Studi Kasus Pada PT.Kalstar Aviation*”

Pihak STMIK Muhammadiyah Jakarta berhak menyimpan mengelola, mendistribusikan, atau mempublikasikan di *internet* atau media lain untuk kepentingan akademis tanpa perlu meminta ijin dari penulis/pencipta karya ilmiah tersebut.

Penulis bersedia menanggung secara pribadi, tanpa melibatkan pihak STMIK Muhammadiyah Jakarta, segala bentuk tuntutan hukum yang timbul atas pelanggaran Hak Cipta dalam karya ilmiah ini.

Demikian pernyataan ini saya buat dengan sebenarnya

Jakarta, 22 November 2017

Yang Menyatakan

Saeful Mahmud

# HALAMAN MOTO

***“Pantang Menyerah Sampai Tujuan Tercapai”***

# KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT yang telah berkenan memberikan rahmat dan hidayah-Nya, sehingga penulisan Skripsi dapat diselesaikan dengan tepat waktu. Penulis menyadari bahwa tanpa bimbingan dan dorongan dari semua pihak, penulisan Skripsi ini tidak dapat terselesaikan dengan baik. Oleh karena itu pada kesempatan ini penulis menyampaikan rasa terima kasih kepada:

1. Bapak Faiz Rafdhi CH, M.Kom selaku Ketua STMIK Muhammadiyah Jakarta.
2. Bapak M. Zen Fitriansyah, S.Kom selaku Wakil Ketua Bidang Akademik.
3. Bapak M. Arief Sutisna, S.Kom selaku Wakil Ketua Bidang Keuangan.
4. Bapak Himawan Dwiatmodjo.SH,LL.MM selaku Wakil Ketua Bidang Kemahasiswaan.
5. Bapak Hendarto, M.Kom selaku Kepala Jurusan Teknik Informatika.
6. Bapak M. Arief Sutisna, S.Kom selaku pembimbing dalam penulisan skripsi ini.
7. Ibu saya dan ayah saya beserta keluarga yang telah memberikan dukungan terbaik bagi penulis.
8. Seluruh staff dan dosen di lingkungan STMIK Muhammadiyah Jakarta.
9. Sahabat, teman-teman dan rekan-rekan mahasiswa STMIK Muhammadiyah Jakarta yang telah memberikan dorongan semangat dalam pembuatan skripsi ini.

Penulis tidak mampu membalas jasa baiknya, hanya berdoa semoga Allah SWT memberikan ganjaran yang setimpal dengan amal perbuatannya. Dan semoga tulisan ini dapat berguna bagi penulis dan pembaca dalam rangka pengembangan ilmu pengetahuan.

Jakarta, 22 November 2017

Saeful Mahmud

# ABSTRAK

Saeful Mahmud, 13.3.0132. **Sistem Informasi Controlling Sertifikasi Awak Pesawat (Studi Kasus PT.Kalstar Aviation)**. Skripsi. Jakarta: Program Studi Teknik Informatika Sekolah Tinggi Manajemen Informatika dan Komputer Muhammadiyah Jakarta. 2017

Sebagai sistem yang berfungsi untuk melakukan manajemen kontrol dari sebuah divisi, aplikasi ini dilengkapi dengan fitur yang memudahkan user dalam melakukan aktifitas. Fungsi utama adalah controlling sertifikasi, yaitu mengontrol kapan sertifikasi akan berakhir dan mengontrol sertifikasi apa saja yang harusnya dimiliki oleh awak pesawat.

Disisi lain aplikasi ini memberikan tambahan fungsi yaitu berupa pengajuan pembaruan sertifikasi secara komputerisasi, monitoring progres, melihat remark dan email gateway yang terkirim secara otomatis. Semua fitur yang ada di rancang sedemikian rupa sehingga membentuk satu kesatuan sistem yang saling berkaitan.

Aktifitas kerja dengan mengandalkan cara-cara lama ataupun administrasi yang kurang rapi bisa mulai ditinggalkan saat sistem ini dihadirkan. User tidak lagi direpotkan dengan mencari data dari tumpukan kertas ataupun membuka file-file spreadsheet untuk menemukan dan mengolah sertifikasi.

Dibuatnya Sistem Informasi Controlling Sertifikasi Awak Pesawat adalah sebagai bentuk upaya dari perusahaan untuk menerapkan efisiensi dari segi waktu, biaya dan tenaga serta meminimalisir kesalahan yang berdampak pada kinerja sumber daya manusia yang dimiliki.

**Kata Kunci: *Sistem Informasi, Awak Pesawat, Controlling Sertifikasi.***

# DAFTAR ISI

[LEMBAR PERSETUJUAN ii](#_Toc498404884)

[LEMBAR PERNYATAAN iii](#_Toc498404885)

[KATA PENGANTAR v](#_Toc498404886)

[ABSTRAK vi](#_Toc498404887)

[DAFTAR ISI vii](#_Toc498404888)

[DAFTAR GAMBAR x](#_Toc498404889)

[DAFTAR TABEL xii](#_Toc498404890)

[BAB I 1](#_Toc498404891)

[1.1 LATAR BELAKANG 1](#_Toc498404892)

[1.2 IDENTIFIKASI MASALAH 2](#_Toc498404893)

[1.3 RUMUSAH MASALAH 2](#_Toc498404894)

[1.4 BATASAN MASALAH 3](#_Toc498404895)

[1.5 SISTEMATIKA PENULISAN 3](#_Toc498404896)

[BAB I Pendahuluan 3](#_Toc498404897)

[BAB II Landasan Teori 3](#_Toc498404898)

[BAB III Metodologi Penelitian 3](#_Toc498404899)

[BAB IV Hasil dan Pembahasan 4](#_Toc498404900)

[BAB V Penutup 4](#_Toc498404901)

[BAB II 5](#_Toc498404902)

[2.1 PENELITIAN SEBELUMNYA 5](#_Toc498404903)

[2.2 PENGERTIAN SISTEM 6](#_Toc498404904)

[2.3 PENGERTIAN INFORMASI 9](#_Toc498404905)

[2.4 SISTEM INFORMASI 10](#_Toc498404906)

[2.5 PENGERTIAN PERANCANGAN SISTEM 11](#_Toc498404907)

[2.6 PENGERTIAN CONTROLLING 11](#_Toc498404908)

[2.7 SISTEM INFORMASI CONTROLLING 12](#_Toc498404909)

[2.7.1 Ciri-ciri Sistem Informasi Controlling Adalah Sebagai Berikut: 12](#_Toc498404910)

[2.8 INTERNET 12](#_Toc498404911)

[2.8.1 Pengertian Internet 12](#_Toc498404912)

[2.8.2 HyperText Transfer Protocol (HTTP) 13](#_Toc498404913)

[2.8.3 Uniform Resource Locator (URL) 13](#_Toc498404914)

[2.8.4 Web Browser 13](#_Toc498404915)

[2.8.5 Email Gateway 14](#_Toc498404916)

[2.9 DATABASE 14](#_Toc498404917)

[2.9.1 Pengertian Database 14](#_Toc498404918)

[2.9.2 Database Management System(DBMS) 14](#_Toc498404919)

[2.10 AWAK PESAWAT 15](#_Toc498404920)

[2.10.1 Pengertian Awak Pesawat 15](#_Toc498404921)

[2.11 SERTIFIKASI 15](#_Toc498404922)

[2.11.1 Pengertian Sertifikasi 15](#_Toc498404923)

[2.11.2 Macam – macam Sertifikasi Awak Pesawat 16](#_Toc498404924)

[2.12 PERANGKAT LUNAK IMPLEMENTASI SISTEM 17](#_Toc498404925)

[2.12.1 Web Application 17](#_Toc498404926)

[2.12.2 Web Server 17](#_Toc498404927)

[2.12.3 PHP 17](#_Toc498404928)

[2.12.4 MySQL 17](#_Toc498404929)

[2.12.5 APACHE 18](#_Toc498404930)

[2.12.6 WAMP 18](#_Toc498404931)

[2.12.7 CI(CodeIgniter) 19](#_Toc498404932)

[2.13 TOOLS PERANCANGAN SISTEM 20](#_Toc498404933)

[2.13.1 UML(Unified Modeling Language) 20](#_Toc498404934)

[BAB III 27](#_Toc498404935)

[3.1 TUJUAN DAN MANFAAT PENELITIAN 27](#_Toc498404936)

[3.1.1 Tujuan Penelitian 27](#_Toc498404937)

[3.1.1 Manfaat Penelitian 27](#_Toc498404938)

[3.2 TEMPAT DAN WAKTU PENELITIAN 27](#_Toc498404939)

[3.2.1 Gambaran Umum Perusahaan 27](#_Toc498404940)

[3.3 TIME SCHEDULE 29](#_Toc498404941)

[3.4 METODE PENGUMPULAN DATA 29](#_Toc498404942)

[3.4.1 Observasi 29](#_Toc498404943)

[3.4.2 Metode Wawancara 30](#_Toc498404944)

[3.4.3 Studi Pustaka 31](#_Toc498404945)

[3.5 METODOLOGI PENELITIAN 31](#_Toc498404946)

[3.5.1 Fountain Model 31](#_Toc498404947)

[3.5.2 Kelebihan dan kekurangan *Fountain* Model 33](#_Toc498404948)

[BAB IV 34](#_Toc498404949)

[4.1 Analisa Sistem (*System Analysis*) 34](#_Toc498404950)

[4.1.1 Analisa Sistem Yang Berjalan 34](#_Toc498404951)

[4.1.2 Analisa Sistem yang diusulkan 36](#_Toc498404952)

[4.2 Spesifikasi Kebutuhan *(Requirements Spesifications)* 37](#_Toc498404953)

[4.2.1 Kebutuhan Hadware 37](#_Toc498404954)

[4.2.2 Kebutuhan Software 37](#_Toc498404955)

[4.2.3 Kebutuhan Fungsional 37](#_Toc498404956)

[4.3 Desain Sistem (System Design) 39](#_Toc498404957)

[4.3.1 Use Case Digram Sistem 39](#_Toc498404958)

[4.3.2 Class Diagram 53](#_Toc498404959)

[4.3.3 Squence Diagram 54](#_Toc498404960)

[4.3.4 Activity Diagram 56](#_Toc498404961)

[4.3.5 Design Database 58](#_Toc498404962)

[4.3.6 Rancangan Interface 64](#_Toc498404963)

[4.4 Testing Aplikasi (Application Testing) 68](#_Toc498404964)

[4.4.1 Rencana Pengujian 68](#_Toc498404965)

[4.4.2 Hasil Pengujian 71](#_Toc498404966)

[4.4.3 Kesimpulan Hasil Pengujian 77](#_Toc498404967)

[4.5 Implementasi (Implementation/Operation) 78](#_Toc498404968)

[4.5.1 Implementasi Perangkat Lunak 78](#_Toc498404969)

[4.5.2 Implementasi Perangkat Keras 78](#_Toc498404970)

[4.5.3 Hasil Implementasi Antar Muka 79](#_Toc498404971)

[4.6 Pemeliharaan dan Pengembangan*(Maintenance and Evolution)* 85](#_Toc498404972)

[BAB V 86](#_Toc498404973)

[5.1 Kesimpulan 86](#_Toc498404974)

[5.2 Saran 86](#_Toc498404975)

# DAFTAR GAMBAR

***Gambar*** *2.1 WAMP 18*

***Gambar*** *2.2 Logo CodeIgniter 18*

***Gambar*** *3.1 Fountain Model 33*

***Gambar****. 4.1 Activity Diagram Sistem berjalan 36*

***Gambar****. 4.2 Activity Diagram Sistem usulan 37*

***Gambar****. 4.3 Use Case Diagram Integrasi Digital 40*

***Gambar****. 4.4 Class Diagram 54*

***Gambar****. 4.5 Sequence diagram staff controlling 55*

***Gambar****. 4.6 Sequence diagram class certifi renewal 56*

***Gambar****. 4.7 Aktifity diagram sistem informasi controlling sertifikasi. 57*

***Gambar****. 4.8 desain output view controlling Expiry. 65*

***Gambar****. 4.9 desain output view list controlling certifi. 66*

***Gambar****. 4.10 desain output view certifi renewal. 66*

***Gambar****. 4.11 desain output view detail certifi renewal. 67*

***Gambar****. 4.12 desain input view add/edit certifi list. 67*

***Gambar****. 4.13 desain input view request renewal, verify dan approval. 68*

**Gambar** *4.14 Halaman view crew profile 80*

***Gambar*** *4.15**Halaman tambah crew profile 80*

**Gambar** *4.16**Halaman view certifi Type 81*

**Gambar** *4.17**Halaman tambah certifi Type 81*

***Gambar*** *4.18 Halaman view user account 82*

***Gambar*** *4.19**Halaman view akses user account 82*

***Gambar*** *4.20**Halaman short view controlling Expiry 83*

***Gambar*** *4.21 Halaman view controlling Expiry 83*

***Gambar*** *4.22 Halaman view certifi list 84*

***Gambar*** *4.23 Halaman view certifi renewal 84*

***Gambar*** *4.24**Halaman view detail certifi renewal 85*

***Gambar*** *4.25 view email otomatis 85*

# DAFTAR TABEL

***Tabel*** *2.1 Simbol Class Diagram 20*

***Tabel*** *2.2 Simbol Componen Diagram 20*

***Tabel*** *2.3 Simbol Deployment Diagram 21*

***Tabel*** *2.4 Simbol Use Case Diagram 22*

***Tabel*** *2.5 Simbol Actifity Diagram 23*

***Tabel*** *2.6 Simbol StateChart Diagram 24*

***Tabel*** *2. 7 Simbol Sequence Diagram 25*

***Tabel 2.****8 Simbol Collaboration Diagram. 25*

***Tabel*** *3.1 Struktur Organisasi Perusahaan 28*

***Tabel*** *3.2 Time Schedule 28*

***Tabel*** *3.3 Pedoman Wawancara 29*

***Tabel*** *3.4 Lima 5 Penelitian Sebelumnya 31*

***Tabel*** *4.1 tpengguna 59*

***Tabel*** *4.2 tmodulakses 59*

***Tabel*** *4.3 tcrewType 60*

***Tabel*** *4.4 tcrewcertifiType 60*

***Tabel*** *4.5 tcertifiType 60*

***Tabel*** *4.6 tcrewprofile 61*

***Tabel*** *4.7 tcontrollingcertifi 62*

***Tabel*** *4.8 tcontrollingExpiry 63*

***Tabel*** *4.9 tcertifirenewal 64*

***Tabel*** *4.10 Pengujian Login 65*

***Tabel*** *4.11 Pengujian Group Account 69*

***Tabel*** *4.12 Pengujian Group Data Suport 70*

***Tabel*** *4.13 Pengujian Group Information 71*

***Tabel*** *4.14 Hasil Pengujian Login/Logout 72*

***Tabel*** *4.15**Hasil Pengujian Group Account 73*

***Tabel*** *4.16**Hasil Pengujian Group Data Support 76*

***Tabel*** *4.17**Hasil Pengujian Group Information 78*

***Tabel*** *4.18**Hasil Pengujian Export dan Terima Email 78*

# BAB I

**PENDAHULUAN**

## LATAR BELAKANG

Menjadi salah satu penerbangan komersial yang mendominasi wilayah kalimantan, PT. Kalstar Aviation*(PT.KSA)* terus meningkatkan service kepada pelangganya agar tetap setia menggunakan jasa angkutan udara yang dimiliki. Transportasi udara mulai menjadi favorit karena dirasa lebih memiliki efisiensi waktu jika jaraknya jauh untuk ditempuh, serta persaingan harga tiket yang semakin murah menjadi salah satu daya tarik konsumen.

Tak lepas dari kata service erat hubunganya dengan layanan yang diberikan oleh para awak pesawat atau yang sering dikenal dengan *aircrew*. Awak pesawat terdiri dari *cabin crew*(Pramugari) dan *cockpit crew*(Pilot)*,* selain bertugas untuk memastikan dan menjaga keselamatan semua penumpang pesawat terbang mulai dari sebelum *take off* sampai dengan selesai *landing*, mereka bisa jadi tolak ukur penilaian penumpang kepada perusahaan. Untuk menjalankan tugas sehari-harinya, awak pesawat membutuhkan lisensi berupa sertifikasi keahlian. Sertifikasi sendiri merupakan hal wajib yang harus dimiliki oleh mereka, maka dari itu perlu adanya controlling terhadap sertifikasi dari masing-masing awak pesawat sehingga mereka tetap bisa menjalankan tugasnya sebaik mungkin.

Adanya sertifikasi dan kewajiban untuk melakukan pembaruan mengharuskan perusahaan melakukan controlling jadwal kapan harus dilakukanya perpanjangan sertifikasi para awak pesawat. DSKU*(Direktorat Sertifikasi Kelaikan Udara)* juga rutin melakukan audit tertib administrasi. Melihat banyaknya sertifikasi yang dimiliki oleh awak pesawat, seringkali mengakibatkan keterlambatan pembaruan sehingga menyebabkan mereka tidak diizinkan melakukan penerbangan bahkan bisa jadi dicabut izin terbangnya oleh pihak DSKU*.* Sebenarnya controlling sudah dijalankan sebagaimana mestinya, akan tetapi cara yang digunakan adalah masih berbentuk elektronik manual yaitu dengan memonitor satu-persatu data yang ada pada aplikasi *ms.excel* tanpa ada *reminder* khusus. Selain itu saat data sudah didapatkan dan diserahkan ke pihak keuangan untuk melakukan persetujuan pembayaran, seringkali pihak *finance* masih harus mencari-cari kebenaran data yang diberikan oleh pihak *controlling* sehingga waktu yang dibutuhkan semakin lama.

Berdasarkan permasalahan yang ada pada pembahasan diatas, maka sangat dibutuhkan sekali sistem informasi yang dapat mempermudah jalanya pekerjaan khususnya untuk pihak *controlling* sertifikasi. Oleh karena itu penelitian ini merencanakan pembuatan *“sistem informasi controlling sertifikasi awak pesawat di PT.KSA”*. sistem ini berbasis web dapat diakses dari manapun dan kapanpun serta yang paling penting dapat memberikan reminder otomatis karena memiliki fitur *message gateway*, sehingga tidak ada alasan untuk terjadi keterlambatan penjadawalan sertifikasi yang menyebabkan awak pesawat dikenai sangsi oleh pihak *DSKU*.

## IDENTIFIKASI MASALAH

Melihat adanya latar belakang diatas, maka identifikasi masalaha dalam penelitian ini adalah sebagai berikut:

1. Sistem informasi controlling sertifikasi awak pesawat di PT.KSA belum terealisasi.
2. Penjadwalan perpanjangan sertfifikasi awak pesawat sering terlambat baik karena terlewatkanya jadwal perpanjangan ataupun lamanya pihak *finance* memverifikasi data.
3. Belum ada sebuah sistem yang otomatis memberikan *reminder* apabila ada sertifikasi yang sudah mendekati kedaluwarsa*/Expiry*

## RUMUSAH MASALAH

Guna memperjelas permasalahan yang timbul, dalam hal ini diformulasikan ke dalam sebuah bentuk rumusan masalah. Adapun rumusan masalahnya:

1. Bagaimana merancang sistem informasi controlling sertifikasi yang dapat memberikan laporan dan *reminder* berupa *message* saat ada yang mendekati jadwal perpanjangan *(expired).*
2. Bagaimana sistem informasi controlling sertifikasi awak pesawat ini bisa diimplementasikan untuk tertib administrasi sehingga tidak ada sangsi lagi dari DSKU *(Direktorat Sertifikasi Kelaikan Udara)?*

## BATASAN MASALAH

Untuk lebih mengarahkan pembuatan rencana skripsi ini dan untuk mempermudah pembahasan maka pembatasan masalahnya adalah:

1. Pembahasan masalah ini hanya dibatasi pada lingkup PT.Kalstar Aviation khususnya di bagian *staff controlling*.
2. Pembahasan ini hanya pada ruang lingkup sertifikasi awak pesawat yaitu Pilot/*cockpit crew* dan Pramugari/*cabin crew.*
3. Untuk memenuhi kebutuhan, perancangan sistem informasi controlling sertifikasi awak pesawat ini dengan bahasa pemrograman PHP dan database menggunakan MySQL.

## SISTEMATIKA PENULISAN

### BAB I Pendahuluan

Pada bab I ini berisi latar belakang masalah, tujuan, batasan masalah, metodologi penelitian, sistematika penulisan dan kerangka pemikiran.

### BAB II Landasan Teori

Bab II dari penulisan buku ini menjelaskan tentang tinjauan pustaka yaitu teori-teori bagaimana tentang sistem informasi. Selain itu juga menjelaskan sertifikasi apa saja yang harus dimiliki oleh *aircrew* untuk menjalankan tugasnya, menjelaskan bagaimana itu metode SDLC Model *Fountain* dan hal lain yang berhubungan dengan penelitian ini.

### BAB III Metodologi Penelitian

Pada Bab III menjelaskan gambaran bagaimana akan dibangun Sistem Informasi Controlling Sertifikasi Awak Pesawat untuk membantu *staff controlling* sertifikasi dalam menjalankan tugasnya dan untuk meminimalisir kesalahan yang akan terjadi di kemudian hari. Perancangan sistem yang digunakan untuk mengolah data, antara lain; APL(Arsitektur Perangkat Lunak), pemodelan system UML *United Model Languange*  diantaranya *actifity diagram, usescase diagram, class diagram, squence diagram* serta pengetahuan management tentang kontroling sertifikasi *aircrew* yang didapat dari hasil wawancara pihak staff controlling.

### BAB IV Hasil dan Pembahasan

Bab ini menjelaskan tentang bagaimana Sistem Informasi Controlling Sertifikasi Awak Pesawat diimplementasikan mulai dengan tahapan pemrograman, spesifikasi perangkat keras dan perangkat lunak, implementasi basis data, proses *input*, *output* serta menu-menu yang mewakili modul yang ada.

### BAB V Penutup

Didalam bab V ini berisi uraian singkat atau kesimpulan dan saran dari penelitian ini.

# BAB II

**LANDASAN TEORI**

## PENGERTIAN SISTEM

Simpulan dari beberapa ahli yang mengerti dengan sistem, bahwasanya sistem adalah prosedur-prosedur yang saling berkaitan membentuk suatu kerangka kerja yang utuh untuk melaksanakan suatu perintah atau pekerjaan sehingga dalam pelaksanaanya antara satu elemen dengan elemen yang lain saling berinteraksi demi mencapai suatu tujuan yang sudah di tentukan.[[1]](#footnote-1)

Sedangkan pandangan beberapa ahli yang digunakan sebagai bahan referensi adalah sebagai berikut:

Menurut Azhar Susanto: **“Sistem adalah kumpulan/group dari sub sistem/bagian/komponen/apapun baik phisik ataupun non phisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai tujuan tertentu”.**[[2]](#footnote-2)

Menurut Sutarman: **“Sistem adalah kumpulan elemen yang saling berinteraksi dalam suatu kesatuan untuk menjalankan suatu proses pencapaian suatu tujuan utama.”**[[3]](#footnote-3)

Dari beberapa pandangan yang ada dari para ahli, ada yang mengatakan bahwa sistem erat kaitanya dengan prosedur. Keduanya tidak bisa dipisahkan satu sama lain. Suatu sistem bisa berjalan apabila di dalamnya terdapat beberapa prosedur yang berjalan.

Menurut Mulyadi: “**Sistem Adalah jaringan prosedur yang dibuat menurut pola yang terpadu untuk melaksanakan kegiatan-kegiatan pokok perusahaan sedangkan prosedur adalah urutan kegiatan klerikal, biasanya melibatkan beberapa orang dalam satu department atau lebih yang dibuat untuk menjamin penanganan secara seragam transaksi perusahaan yang terjadi secara berulang-ulang.”[[4]](#footnote-4)**

Dari pendapat para ahli diatas dapat diambil kesimpulan bahwa sebenarnya sistem adalah sekelompok unsur yang hubunganya saling berkaitan satu dengan yang lain, difungsikan bersama demi mencapai suatu tujuan tertentu. Maka dengan kesimpulan diatas dapat dirinci dalam pengertian sistem yang lebih global sebagai berikut:[[5]](#footnote-5)

1. Setiap sistem terdiri dari unsur-unsur
2. Dari unsur-unsur yang merupakan begian sistem yang terpadu dan saling bersangkutan.
3. Unsur sistem yang saling bersangkutan bekerja sama untuk mencapai tujuan tertentu.
4. Suatu sistem merupakan bagian dari sistem yang lain yang lebih besar.

Suatu Sistem baru bisa dikatakan sebuah sistem informasi apabila didalamnya mengandung dan memenuhi karakteristik utama dari sebuah sistem informasi. Karakteristik utama tersebut menunjukan bahwa sistem yang dibuat benar-benar memberikan solusi dari informasi awal yang didapatkan dalam sistem tersebut dengan alur mulai dari host sampai dengan user atau penerimanya. Beberapa karakteristik sistem yang ada sebagai berikut:

1. Komponen Sistem*(Component)*

Suatu sistem yang ada, terdiri dari kumpulan-kumpulan komponen yang saling berinteraksi. Setiap komponen yang ada berfungsi untuk menjalankan fungsi-fungsi tertentu dari sebuah sistem. Komponen tersebut sering kita kenal dengan sebutan subsistem. Apabila salah satu komponen tersebut tidak bekerja sebagaimana mestinya, maka sistem yang ada tidak dapat bekerja secara optimal.

1. Batasan Sistem*(Boundary)*

Karakteristik sistem yang selanjutnya adalah bahwa sistem informasi memiliki batasan, maksudnya disini sistem informasi harus memiliki pembatas dengan sistem informasi yang lain. Sehingga tugas dari masing-masing sistem terbagi secara jelas, tidak saling tumpang tindih dan tidak saling mengacaukan peranan sistem yang lain. Hal tersebut karena ruang lingkup sebuah sistem itu sudah ada pembagian tersendiri, tidak bisa dicampuradukkan.

1. Lingkungan Luar Sistem *( Environment )*

Selanjutnya, karakteristik dari sistem informasi bahwa sebuah sistem informasi haruslah memiliki lingkungan luar. Environment bisa dikatakan keseluruhan sistem dan juga lingkungan yang berada diluar batasan atau boundary dari sebuah sistem informasi. Tanpa adanya environment, sistem yang ada tidak akan berarti apa-apa.

1. Pengubung Sistem*(Interface)*

Suatu interface atau antarmuka, merupakan karakteristik sebuah sistem yang harus dimiliki. Karakteristik tersebut mengacu pada user yang nanti akan menggunakan sistem tersebut bahwa tujuan dibuatnya sistem adalah agar user yang awam sekalipun dapat dengan mudah mengoperasikanya. Interface juga bisa dikatakan sebuah komponen khusus yang ada pada sistem, apabila interface suatu sistem tidak ada, maka sistem informasi tidak dapat berjalan secara optimal.

1. Masukan sistem *(Input)*

Karakteristik selanjutnya adalah input, bisa dikatakan suatu energy yang dimasukan kedalam sistem. Masukan tersebut dapat berupa masukan perawatan *(maintenance)* agar sistem dapat beroperasi dan masukan signal *(signal input)* yaitu energy yang diproses melalui sistem informasi menuju output.

1. Pengolahan Sistem *(Proses)*

Pengolahan sistem merupakan karakteristik dari sistem informasi yang memiliki tugas utama untuk memproses masukan sistem. Lebih jelasnya pengolahan sistem ini membantu proses pengolahan data secara keseluruhan yang ada sistem informasi, lalu mengirimkan hasil olahan data tersebut menuju output dan selanjutnya informasi yang ada dapat diterima oleh user/pengguna.

1. Keluaran Sistem *(Output)*

Keluaran sistem merupakan energy dari masukan sistem yang sudah diolah oleh pengolahan sistem yang selanjutnya hasil dari pengolahan tersebut dapat dimanfaatkan untuk masukan bagi subsistem yang lain, ataupun untuk user/pengguna.

1. Sasaran Sistem *(Objektive)*

Karakteristik dari sistem yang terakhir adalah objektive/sasaran sistem. Suatu sistem bisa dikatakan berhasil apabila mengenai sasaran yang dituju. Sasaran dari sistem dapat juga dikatakan sebuah analisis berupa siapa saja yang akan menggunakan sistem tersebut. Misalkan sistem informasi yang akan dijalankan adalah para staff pergudangan yang merekap data-data barang maka sistem informasi yang digunakan oleh mereka adalah sistem informasi inventori.

## PENGERTIAN INFORMASI

Informasi adalah suatu hasil pengolahan berasal dari data yang sudah dimanipulasi dan dirangkum sedemikian rupa dengan bentuk dan cara yang lain dari pada bentuk aslinya sebagai tujuan untuk mendapatkan hasil akhir sebagai sarana pengambilan keputusan[[6]](#footnote-6). Pandangan beberapa ahli juga beragam akan tetapi merujuk pada hal yang sama.

Menurut Jogianto : **“Informasi adalah data yang diolah menjadi bentuk yang lebih berguna bagi yang menerimanya”** [[7]](#footnote-7)

Menurut Azhar Susanto : **“Informasi adalah hasil pengolahan data yang memberikan arti dan manfaat”**[[8]](#footnote-8)

Sesuai dengan pandangan para ahli diatas, dapat disimpulkan kembali bahwa sebenarnya informasi adalah sekumpulan data yang sudah diolah sedemikian rupa, sehingga dapat digunakan bagi yang membutuhkanya baik nanti digunakan sebagai sarana pengambilan keputusan ataupun juga sebagai sarana pengetahuan.

Terlepas dari pandangan para ahli diatas tentang pengertian informasi, kualitas dari informasi pun juga perlu diperhatikan. Berikut kualitas dari informasi yang umumnya digunakan sebagai acuan:

1. **Akurat,** akurat disini maksudnya adalah informasi yang di dapat harus bebas dari kesalahan, tidak menyesatkan, tidak ada perkiraan atas informasi yang di dapat. Maksud dari perkiraan diatas adalah, misal seseorang membutuhkan informasi atas jawaban dari pertanyaanya dan membutuhkan satu jawaban tepat, maka hasil yang di dapatkan juga harus berupa jawaban yang tepat bukan jawaban kurang dari nilai yang diharapkan atau lebih dari itu.
2. **Tepat Waktu,** Setiap informasi yang diinginkan, biasanya berhubungan dengan ketepatan waktu yang ada. Sebuah informasi yang diterima melebihi dari waktu yang ditentukan biasanya akan menjadi sebuah informasi yang kedaluarsa dan tidak berguna sehhingga menyebabkan operasional yang ada terganggu. Sehingga diperlukanlah ketepatan waktu atas jawaban dari setiap permintaan informasi yang ada.
3. **Relevan,** relevan disini maksudnya adalah informasi yang diberikan kepada user yang sedang mencari informasi adalah benar-benar informasi yang dibutuhkan. Jadi misalkan seorang akuntan sedang mencari informasi tentang bagaimana rumus neraca, maka yang ditampilkan adalah rumus dari neraca yang dimaksud, bukan rumus persamaan pythagoras atau rumus persamaan atom kimia[[9]](#footnote-9).

## SISTEM INFORMASI

Sistem Informasi adalah suatu sistem komputer yang menyediakan informasi berdasarkan olahan data yang diterima melaluli input – proses - output kepada user dimana user itu sendiri membutuhkan informasi tersebut untuk membantunya dalam mengambil keputusan. Sistem informasi sendiri adalah sebuah kerangka dimana untuk mendapatkan sebuah informasi yang dibutuhkan maka perlu mengkoordinasikan antara sistem komputer dengan manusia.

Sistem Informasi sendiri memiliki fungsi yang berbeda, dalam pengembanganya, sistem informasi punya tujuan sesuai dengan kebutuhan masing-masing user yang akan menggunakan. Adapun dalam penerapanya, sistem yang sudah berjalan tidak bisa berdiri sendiri tanpa menggunakan basis data untuk sarana menampung informasi.

## PENGERTIAN PERANCANGAN SISTEM

Perancangan sistem bisa dikatakan suatu kegiatan untuk melakukan perancangan, penggambaran, dan pendesainan baik untuk pembuatan sistem baru ataupun menggantikan sistem yang secara sebagian ataupun keseluruhan.

## PENGERTIAN CONTROLLING

Controlling atau dalam bahasa indonesia sering kita kenal dengan sebutan pengawasan. Dari satu kata tersebut kita bisa terjemahkan secara garis besar bahwa pengawasan adalah suatu kegiatan melihat, ataupun mengamati sebuah proses yang berjalan. Akan tetapi pengertian dari diri pribadi tidak bisa digunakan sebagai acuan sebuah karya ilmiah. Setiap ahli memiliki ungkapan yang berbeda-beda dalam menerangkan apa itu arti dari controlling, akan tetapi pengertian tersebut tetap berujung pada satu kesimpulan.

Berikut pengertian controlling/pengawasan menurut para ahli:[[10]](#footnote-10)

Menurut Sarworto: **“Pengawasan ialah kegiatan dari manajer yang mengusahakan supaya pekerjaan-pekerjaan dapat terlaksana sesuai dengan rencana yang telah ditetapkan ataupun hasil yang sudah dikehendaki”**

Di dalam definisi yang diungkapkan oleh sarwoto, yang bertindak sebagai seorang controlling/pengawasan adalah seorang manager. Manager dianggap orang mengerti seluk beluk suatu pekerjaan yang berhubungan dengan pengawasan.

Menurut Soekarno K: **“Pengawasan ialah suatu proses yang menentukan mengenai apa yang harus dikerjakan, supaya apa yang harus dikerjakan, supaya apa yang diselenggarakan dapat sejalan sesuai dengan rencana.”**

Dalam hal ini Soekarno K lebih menekankan bahwa pengawasan ialah sebuah proses penentu mengenai suatu hal yang seharusnya dilakukan agar apa yang ingin diselenggerakan berjalan sesuai dengan rencana.

Menurut S.P.Siagian: **“Pengawasan merupakan sebuah proses pengamatan pelaksanaan dari seluruh kegiatan organisasi untuk dapat menjamin sepaya seluruh pekerjaan yang sedang dilakukan dapat sesuai dengan rencana yang sudah ditentukan sebelumnya.”**

Pandangan yang disampaikan oleh S.P.Siagian merupakan salah satu pandangan yang mudah untuk kita uraikan. Dimana proses pengawasan itu bisa dilakukan hanya pada pekerjaan yang tengah berjalan. Bukan pada pekerjaan yang sudah selesai dilaksanakan.

Menurut M.Manullang: **“Pengawasan ialah suatu proses untuk dapat menetapkan pekerjaan apa yang telah dilaksanakan, menilainya, dan juga mengoreksinya. Dan bila perlu dengan sebuah maksud agar pelaksanaan pekerjaan sesuai dengan rencana yang semula.”**

Menurut Henry Fayol: **“Pengawasan ialah terdiri dari pengujian apakah seluruh sesuatu telah berlangsung sesuai dengan rencana yang sudah ditentukan dengan instruksi yang sudah digariskan.”[[11]](#footnote-11)**

## SISTEM INFORMASI CONTROLLING

Sistem Informasi Controlling adalah suatu penggabungan antara sistem informasi dengan kegiatan controlling yang mana fungsi dari sistem tersebut adalah untuk melakukan pengawasan agar pekerjaan yang sudah diplaningkan dapat berjalan sebagaimana mestinya. Sistem informasi controlling telah banyak digunakan untuk kepentingan manajemen kontrol di perusahaan-perusahaan besar. Sering juga sistem tersebut menjadi bagian sub sistem dari sistem induk yang lebih besar seperti halnya sistem inventory, sistem penjualan dan sistem personalia. Adapun yang menjadi objek controlling dari sistem tersebut antara lain stok barang, masa kedaluarsa produk, masa kontrak karyawan, masa aktif sertifikasi, dsb.

### Ciri-ciri Sistem Informasi Controlling

Adapun ciri-ciri dari Sistem Informasi Controlling adalah sebagai berikut:

1. Dalam satu sistem bisa mengontrol beberapa data.
2. Dapat memberikan kesimpulan data secara akurat sesuai dengan keputusan akhir yang sudah ditentukan
3. Dirancang untuk dapat dikembangkan secara tahap demi tahap
4. Berdasarkan pada kaidah/ rule tertentu
5. Keluaranya bersifat informasi

## INTERNET

### Pengertian Internet

Berasal dari dua kata yaitu *interconnection network* adalah jaringan komputer yang saling terhubung antara satu dengan yang lain baik melalui media kabel LAN, *Wifi*, ataupun Satelit sehingga membentuk sistem jaringan mencakup seluruh dunia *(Global Network).*

### HyperText Transfer Protocol (HTTP)[[12]](#footnote-12)

Adalah protokol standar sampai saat ini yang harus diikuti oleh *web browser* untuk meminta dokumen atau halaman serta untuk *web server* dalam menyediakan permintaan yang datang dari browser.

### Uniform Resource Locator (URL)[[13]](#footnote-13)

*Uniform Resource Locator* adalah alamat yang menentukan lokasi suatu file atau halaman yang akan dituju pada suatu *web server.* Adapun URL yang ada saat ini terbagi menjadi beberapa bagian seperti berikut:

1. Protokol yang akan digunakan oleh *web browser* untuk mengambil informasi pada *web server*.
2. Nama komputer/server tempat dimana halaman/informasi itu disimpan.
3. Serta nama file informasi yang sedang kita akses.

Misalkan kita sedang mengakses aplikasi sistem controlling sertifikasi awak pesawat ini pada browser dengan alamat <http://182.172.61.03/home> maka jika dipilah satu persatu bagian yaitu:

Nama Protokolnya : http://

Nama Servernya : 182.172.61.03

Nama Filenya : home

Namun seringkali untuk mempermudah pencarian alamat pada internet biasanya nama server diubah dengan hostname sehingga menjadi <http://www.scsa-kalstar.com/home> .

### Web Browser

*Web Browser* adalah suatu sistem aplikasi yang digunakan di komputer dirancang untuk me-*request* serta menampilkan informasi-informasi dari server.

Contoh web browser yang sering kita temui pada komputer adalah:

* Mozilla Firefox
* Google Crome
* Safari
* Internet Explorer

### Email *Gateway*

*Email* *gateway* adalah sebuah sistem yang digunakan untuk mengirim ataupun menerima *Email* yang mana di dalam sistem tersebut terdapat konfigurasi yang menjembatani antara sistem *email* dengan aplikasi yang akan diintegrasikan dengan *email* tersebut.

## DATABASE

### Pengertian *Database*

Secara umum, *database* atau dalam bahasa indonesia yaitu basis data adalah koleksi data yang saling berkaitan. Dengan kata lain database dapat dianggap sebagai suatu data yang disusun dan terstruktur disimpan dalam media pengingat *(hard disk)* yang tujuanya adalah agar data yang disimpan dapat diakses dengan mudah dan cepat.[[14]](#footnote-14)

### Database Management System(DBMS)[[15]](#footnote-15)

*Database Management System*  atau yang sering disingkat dengan DBMS adalah sebuah software yang digunakan untuk mengelola database, dalam hal ini mulai dari membuat database itu sendiri kemudian membuat tabel serta field-field yang dibutuhkan, melakukan proses-proses yang berlaku setelah database itu dibuat seperti *insert, edit, delete* atau melakukan *query* terhadap data yang sebelumnya sudah dibuat*.*

Ada beberapa macam database seperti database *hirarkis*, jaringan dan *relasional* akan tetapi dalam penelitian ini hanya akan membahas dan menggunakan database *relasional* karena dianggap database tersebut adalah yang paling cocok untuk digunakan dalam penulisan. Sebuah database *relasional* / RDBMS tersusun atas sejumlah tabel yang memiliki hubungan dengan tabel lainya.[[16]](#footnote-16)

## AWAK PESAWAT

### Pengertian Awak Pesawat

Awak Pesawat, atau yang sering dikenal dalam dunia penerbangan *aircrew* adalah dua atau sekelompok orang yang terlatih dan memenuhi persyaratan yang dibentuk sebagai team yang mengoperasikan pesawat terbang. *Aircrew* sendiri disini dibagi menjadi dua yaitu[[17]](#footnote-17):

* *Cockpit Crew* adalah orang yang bertugas di *Cockpit* antara lain:
* *Pilot In Comand* (PIC) sering dikenal dengan kapten penerbangan dan bertindak sebagai pimpinan dalam penerbangan.
* *First Officer* (FO) sering di kenal dengan nama Co-Pilot adalah yang bertugas sebagai asisten dari PIC dalam penerbangan.
* *Flight Enginer* (FE) adalah enginer/teknisi pesawat yang ditugaskan untuk mendampingi saat penerbangan berlangsung.
* *Cabin Crew* adalah orang yang bertugas di cabin pesawat untuk memberikan pengarahan keselamatan serta pelayanan kepada penumpang selama penerbangan. *Cabin Crew* dalam dunia penerbangan disebut juga *Flight Attendant* (FA) atau sering kita kenal Sebagai Pramugari untuk sebutan FA perempuan dan Pramugara untuk sebutan FA Laki-laki.

## SERTIFIKASI

### Pengertian Sertifikasi

Sertifikasi adalah suatu penetapan yang diberikan oleh badan, organisasi ataupun sejenisnya secara profesional terhadap seseorang untuk menunjukkan bahwa orang tersebut mampu dan memenuhi kualifikasi untuk melakukan pekerjaan sesuai dengan sertifikasi yang diberikan. Sertifikasi sendiri biasanya memiliki batas waktu yang sudah ditentukan sehingga orang yang memiliki sertifikasi harus memperbaruinya secara berkala.

### Macam – macam Sertifikasi Awak Pesawat

Untuk satu orang *cockpit crew* baik itu pilot ataupun co-pilot untuk bisa menjalankan kewajibanya paling tidak memilik 14 sertifikasi[[18]](#footnote-18), sedangkan untuk *cabin crew* adalah 9 sertifikasi[[19]](#footnote-19) seperti dibawah ini dan harus dalam keadaan aktif diantaranya:

* *Medical Check*

Cek kesehatan

* *Dangerous Goods*

Sertifikasi tentang barang berat yang berbahaya

* *Aviation Scurity*

Sertifikasi keamanan penerbangan

* *Safety Management System*

Sertifikasi Memangemen keamanan

* *Crew Resort Management*

Sertifikasi Koordinasi Antar Crew

* *Wet Drill Training*

*Training Penyelamatan dalam air*

* *Instrument Rating*

Training alat-alat dalam penerbangan

* *Proficiency Check*

Check Kemampuan menjalankan tugas

* *Approach And Landing Accident Reduction*

Check kemampuan landing kecelakaan

* *Windshear Training*

*Pelatihan memahami pergerakan angin*

* *Line Check Training*
* *Rec Type Rating*
* *EPC*
* *Raduce Vertical Sparation Minimum*
* *Competency Check*
* *Recurrent Rating*
* *Company Checker*

## PERANGKAT LUNAK IMPLEMENTASI SISTEM

### Web Application

*Web Application* / Aplikasi Web adalah suatu perangkat lunak yang dikembangkan untuk mempermudah dari sisi developer ataupun dari sisi user dimana penggunaanya cukup menggunakan web browser tanpa harus menginstal satu persatu di masing-masing pc. Selain itu aplikasi web bisa diakses dari mana saja selama aplikasi tersebut terkoneksi dengan layanan internet baik melalui *operating system* yang sama ataupun berbeda dengan kata lain web aplikasi mendukung sistem operasi *cross platform*.

### Web Server[[20]](#footnote-20)

*Web Server* adalah server yang digunakan dan mampu untuk melayani koneksi transfer antar data di dalam protokol HTTP. *Web Server* adalah suatu hal yang wajib dimiliki dalam pembuatan aplikasi berbasis web karena bertugas menjalankan alur proses sistem yang berjalan.

### PHP[[21]](#footnote-21)

PHP *(Hypertext Preprocessor)* adalah bahasa pemrograman open source dibawah lisense *PHP License* dikembangkan pada tahun 1995 oleh Rasmus Lerdorf yang dapat dipergunakan oleh semua orang secara gratis tanpa harus membayar, dirancang sebagai bahasa pemrograman *script server-side* yang didesain untuk pengembangan aplikasi berbasis web. Disebut sebagai bahasa pemrograman *server-side* karena PHP diproses pada komputer utama yaitu server berbeda dengan bahasa pemrograman lain seperti javascript atau turunanya yang pemrosessanya dilakukan pada web browser masing-masing client.

### MySQL[[22]](#footnote-22)

MySQL(dibaca: *mai-se-kyu-el*) adalah software yang tergolong sebagai DBMS *(Database Management System)* yang bersifat *Open Source* yang sudah mendukung manajemen basis data relasional RDBMS dibawah lisence GPL*(General Public Lisence).* Dalam keteranganya *Open Source* menyatakan bahwa software ini dilengkapi dengan *source code*, artinya kode yang dipakai untuk membuat MySQL itu sendiri disertakan dalam paket tersebut, selain itu *executeble*-nya atau kodenya yang dapat dijalankan di semua sistem operasi yang dapat diperoleh dengan cara mendownload langsung dari internet secara gratis.

Sebagai software DBMS yang sudah mendukung relasional MySQL memiliki sejumlah fitur diantaranya:

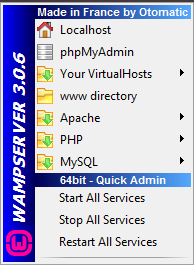
* *Multiplatform*
* Andal, cepat, dan mudah digunakan.
* Jaminan Keamanan Akses
* Mendukung Banyak Bahasa Pemrograman
* Dukungan SQL

### APACHE

Apache adalah salah satu dari sekian *web server* yang bersifat open source yang sudah digunakan oleh banyak *developer* dalam mengembangkan programnya baik yang *open source* ataupun yang berbayar. Apache saat ini menjadi *web server* paling populer di kalangan *programmer* karena kemudahan penggunaanya dan support hampir di semua sistem operasi.

### WAMP

Wamp adalah sebuah paket aplikasi server yang berada di *localhost* dimana untuk penginstalanya dikususkan hanya untuk sistem operasi windows. WAMP sendiri adalah singkatan dari paket-paket yang tersedia dalam aplikasi tersebut seperti:

* W = Windows maksudnya aplikasi ini mendukung hanya untuk sistem operasi windows.
* A = Apache artinya paket aplikasi WAMP ini didalamnya sudah tersedia apache server.
* M = MySQL dimana aplikasi WAMP ini sudah mendukung Database Management System yaitu MySQL.
* P = PHP dimana aplikasi ini support untuk aplikasi web yang menggunakan bahasa pemrograman berbasis PHP.

***Gambar 2.1*** *WAMP*

### CI(CodeIgniter)*[[23]](#footnote-23)*

*CodeIginiter* disingkat CI adalah *framework* pengembangan aplikasi *(Application Development Framework)* dengan menggunakan PHP, bisa juga dikatakan adalah suatu kerangka untuk bekerja dan membuat program dengan menggunakan PHP yang lebih sistematis. Pemrograman tidak perlu membuat program dari awal *(frow scratch)*, karena Codeigniter sudah menyediakan sekumpulann fungsi-fungsi yang diperlukan untuk menyelesaikan pekerjaan yang umumnya dilakukan.



***GAMBAR 2.2*** *Logo CodeIgniter*

## TOOLS PERANCANGAN SISTEM

### UML*(Unified Modeling Language)[[24]](#footnote-24)*

UML merupakan suatu metode untuk memvisualisasikan perancangan sistem yang berorientasi objek, UML sendiri menjadi bahasa standarisasi visualisasi agar orang dari belahan dunia yang lain juga bisa mengerti rancangan yang kita buat. Bisa dikatakan UML adalah sebagai rancangan dasar suatu program*(penggambaran rancangan)* untuk mendapatkan hasil akhir dari suatu program secara maksimal.

Ada beberapa macam jenis UML yang ada:

#### Class Diagram[[25]](#footnote-25)

Tujuan utama dari diagram kelas adalah untuk menciptakan sebuah kosa kata yang digunakan untuk menuangkan pemikiran analys agar dapat diterima oleh pengguna. Class Diagram biasanya merupakan ide-ide atau konsep yang ada dalam aplikasi. Misalnya jika saat ini sedang membuat aplikasi controlling sertifikasi maka maka diagram kelas mewakili seperti *crew profile, Type certifi, crew certifi* dan *Expiry*.

|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **GAMBAR** | **NAMA** | **KETERANGAN** |
| 1 |  | *Generalization* | Dimana hubungan antara object anak*(descendent)* berbagi perilaku dan struktur dari object induk*(ancestor)* yang ada diatasnya |
| 2 |  | *Nary Association* | Upaya untuk menghindari asosiasi dengan lebih dari 2 objek |
| 3 |  | *Class* | Himpunan dari objek-obkjek yang berbagi atribut operasi yang sama. |
| 4 |  | *Collaboration* | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor. |
| 5 |  | *Realization* | Operasi yang benar-benar dilakukan oleh suatu objek. |
| 6 |  | *Dependency* | Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri(independent) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri. |
| 7 |  | *Association* | Apa yang menghubungkan antara objek satu dengan objek lainya. |

***Tabel*** *2.1 Simbol Class Diagram*

#### Component Diagram[[26]](#footnote-26)

Diagram Komponen adalah diagram yang didalamnya lebih berfokus pada komponen yang ada pada system dan hubungan antara komponen tersebut. Diagram Komponen juga dapat berguna dalam merancang sistem berbasis komponen karena berfokus pada orientasi object dan desain.

|  |  |  |
| --- | --- | --- |
| **SIMBOL** | **NAMA** | **KETERANGAN** |
| Generic Component | Komponen | Menggambarkan sebuah entitas software dalam sebuah sistem. |
|  | *Depedency* | Sebuah dependency digunakan untuk menotasikan relasi antara dua komponen. |

***Tabel*** *2.2 Simbol Componen Diagram*

#### Deployment Diagram[[27]](#footnote-27)

Menggambarkan suatu detail bagaiman komponen di deploy dalam infrastruktur sistem. Misalkan dalam sebuah aplikasi web memiliki sebuah node perangkat keras dengan fungsi sebagai web server dan server database, kemudian yang berjalan dalam setiap node adalah aplikasi web dan database.

|  |  |  |
| --- | --- | --- |
| **SIMBOL** | **NAMA** | **KETERANGAN** |
| Generic Component | Komponen | Pada deployment diagram, komponen-komponen yang ada diletakkan di dalam node untuk memastikan keberadaan posisi mereka. |
| C:\Users\Windows 8.1\Downloads\6-DEPLOYMENT.jpg | *Node* | Node menggambarkan bagian-bagian hardware dalam sebuah sistem. Notasi untuk node diambarkan seagai kubus 3 dimensi.. |
|  | *Association* | Sebuah association digambarkan sebagai sebuah garis yang mengubungkan dua node yang mengindikasikan jalur komunikasi antara element-element hardware. |

***Tabel*** *2.3 Simbol Deployment Diagram*

#### Use Case Diagram[[28]](#footnote-28)

*Use case* diagram dapat dikatakan adalah sebuah abstraksi hubungan antara system dengan aktor yang menggunakan sistem itu sendiri. *Use case* bekerja dengan menjalankan tipe interaksi antara user dengan sebuah sistem tentang bagaimana sistem tersebut dipakai.

|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **GAMBAR** | **NAMA** | **KETERANGAN** |
| 1 |  | *Actor* | Menspesifikasikan himpunan peran yang pengguna gunakan ketika berinteraksi dengan sistem. |
| 2 |  | *Dependency* | Hubungan dimana perubahan terjadi pada suatu element yang mandiri*(independent)* akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri *(not independent)* |
| 3 |  | *Generalization* | Hubungan dimana object anak *(descendent)* berbagi perilaku dan struktur data objeck yang ada di atasnya objek induk *(ancestor).* |
| 4 |  | *Include* | Menspesifikasikan bahwa *use case* sumber secara *eksplisit.* |
| 5 |  | *Extend* | Menspesifikasikan bahwa *use case* target memperluas perilaku dari *use case* sumber pada suatu titik yang diberikan. |
| 6 |  | *Association* | Apa yang menghubungkan antara objek satu dengan objek lainya. |
| 7 |  | *System* | Menspesifikasikan paket yang menampilkan sistem secara terbatas. |
| 8 |  | *Use Case* | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil terukur suatu aktor. |
| 9 |  | *Collaboration* | Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebh besar dari jumlah dan elemen-elemenya(sinergi) |
| 10 |  | *Note* | Elemen Fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi. |

***Tabel*** *2.4 Simbol Use Case Diagram*

#### Activity Diagram[[29]](#footnote-29)

Diagram aktifitas adalah sebuah diagram yang menggambarkan bagaimana alur suatu sistem yang dirancang itu berjalan. Dalam diagram ini di sebutkan bagaimana awal sistem mulai bekerja dan berakhirnya sistem, selain itu dalam aktifity diagram memungkinkan terjadinya alur sistem yang pararel.

|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **GAMBAR** | **NAMA** | **KETERANGAN** |
| 1 |  | *Fork* | Percabangan |
| 2 |  | *Join* | Penggabungan |
| 2 |  | *Action* | State dari sistem yang mencerminkan eksekusi dari suatu aksi. |
| 3 |  | *Initial Node* | Bagaimana objek dibentuk atau diawali. |
| 4 |  | *Actifity Final Node* | Bagaimana objek dibentuk dan diakhiri. |
| 5 |  | *Decission* | Digunakan untuk menggambarkan suatu keputusan / tindakan yang harus diambil pada kondisi tertentu. |
| 6 |  | *Line Conector* | Digunakan untuk menghubungkan satu simbol ke simbol yang lain |

***Tabel*** *2.5 Simbol Actifity Diagram*

#### Statechart Diagram[[30]](#footnote-30)

Adalah suatu diagram yang menggambarkan bagaimana suatu sistem berjalan mulai dari awal sistem diinisialisasi, sampai dengan di akhiri. Selain itu statechart diagram juga menggambarkan transisi dan perubahan keadaan dari state satu ke state yang lain.

|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **GAMBAR** | **NAMA** | **KETERANGAN** |
| 1 |  | *State* | Nilai atribut dan nilai link pada suatu waktu tertentu yang dimiliki oleh objeck. |
| 2 |  | *Initial Pseudo State* | Bagamana objek dibentuk atau diawali. |
| 3 |  | *Final State* | Bagaimana Objek dibentuk dan diakhiri. |
| 4 |  | *Transition* | Sebuah kejadian yang memicu sebuah state objek dengan cara memperbaharui stau atau lebih nilai atributnya. |
| 5 |  | *Association* | Apa yang menghubungkan antara objek satu dengan objek lainya. |
| 6 | ***Tabel*** *2. 6 Simbol StateChart Diagram* | *Node* | Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi. |

#### Sequence Diagram[[31]](#footnote-31)

*Squence Diagram* atau bisa disebut sebagai diagram urutan adalah suatu diagram yang menggambarkan skenario program berjalan dimulai dari level yang paling rendah hingga sampai ke level yang paling tinggi dalam sebuah sistem secara berurutan.

|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **GAMBAR** | **NAMA** | **KETERANGAN** |
| 1 |  | *LifeLine* | Objek entity, antarmuka yang saling berinteraksi. |
| 2 |  | *Message* | Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi. |
| 3 |  | *Message*  ***Tabel 2.****7. Simbol Sequence Diagram* | Spesifikasi dari komunikasi antar objek yang membuat informasi-informasi tentang aktifitas yang terjadi. |

#### Collaboration Diagram[[32]](#footnote-32)

Collaboration diagram adalah diagram yang lebih menekankan pada peran masing-masing objek bukan pada waktu pesan disampaikan.

|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **SIMBOL** | **NAMA** | **KETERANGAN** |
| 1 | ; Object!  **; Object1** | *Object* | Object merupakan instance dari sebuah class dan dituliskan tersusun secara hirzontal. Digambarkan sebagai sebuah class (kotak) dengan nama obyek di dalamnya yang diawali dengan sebuah titik koma. |
| 2 |  | *Actor* | Actor juga dapat berkomunikasi dengan object, maka actor juga daoat diurutkan sebagai kolom. Simbol actor sama dengan simbol pada Actor Use Case Diagram. |
| 3 | message | *Message* | Message, digambarkan dengan anak oanah yang mengarah antar obyek dan diberi label urutan nomor yang mengindikasikan urutan komunikasi yang terjadi antar obyek. |

***Tabel*** *2.8 Simbol Collaboration Diagram*

# BAB III

**METODOLOGI PENELITIAN**

## TUJUAN DAN MANFAAT PENELITIAN

### Tujuan Penelitian

Tujuan dari dilakukanya penelitian serta pembuatan sistem informasi controlling sertifikasi awak pesawat ini adalah:

1. Menciptakan rancangan secara terintegrasi sistem informasi controlling sertifikasi awak pesawat dengan menggunakan rekaya sistem metode SDLC *(System Development Life Cycle)* Model *Fountain* dan perangkat permodelan UML *(Unified Modelling Language).*
2. Mengimplementasikan serta menerapkan sistem yang sebelumnya dibuat kedalam sistem aplikasi menggunakan bahasa pemrograman PHP *framework* *codeigniter* dan menggunakan *database* MySQL.

### Manfaat Penelitian

Sedangkan manfaat yang akan didapatkan dari penelitian ini antara lain:

1. Memberikan kontribusi terhadap ilmu pengetahuan khususnya pengetahuan sistem informasi akan pengetahuan-pengetahuan baru yang belum diketahui sebelumnya.
2. Membantu pihak perusahaan yang digunakan untuk penilitian sebagai solusi masalah jangka panjang serta membantu pihak user yang berhubungan langsung dengan masalah tersebut.

## TEMPAT DAN WAKTU PENELITIAN

Tempat penelitian dilaksanakan pada PT.Kalstar Aviation yang beralamat di Villa Melati Mas Blok SR 1 No.14 BSD Tangerang. Untuk penelitian dilakukan dari bulan agustus sampai dengan september 2017.

### Gambaran Umum Perusahaan

#### Sejarah

Kalstar Aviation adalah maskapai penerbangan yang berkantor pusat di BSD Serpong Tangerang Selatan yang terdaftar dalam Otoritas Penerbangan Sipil Indonesia dengan Kategori 1 untuk kualitas keselamatan penerbangan.

Didirikan Oleh Andi Masyhur pada tahun 2000 Kalstar Aviation adalah anak perusahaan dari Kalstar Nusantara yang juga menaungi beberapa perusahaan antara lain Star Cargo, Kalstar Tour & Travel dan Star Reksa Dana. Nama Kalstar Sendiri diambil dari basis rute penerbangan yang dilaluinya yang sebagian besar yaitu kalimantan sehingga dinamakan Kal yang berarti Kalimantan dan Star yaitu Bintang jadi bisa dikatakan di wilayah kalimantan Kalstar merupakan bintang dari berbagai perusahaan penerbangan yang ada.

#### Struktur Organisasi dan Fungsi

Struktur Organisasi yang ada pada PT Kalstar Aviation merupakan struktur terbaru dimana struktur tersebut hanya diambil pada level manager ke atas.

|  |  |
| --- | --- |
| KOMISARIS | Erizal Gani |
|  | Indra Setiawan |
| PRESIDENT DIREKTUR | Andi Masyhur |
| VICE PRESIDEN | Abdul Haris Mulyana |
| DIREKTUR KEUANGAN & ADMINISTRASI | Yefrizal |
| DIREKTUR KOMERSIAL | I Wayan Parka |
| MANAGER SAFETY | Abdul Kadir |
| MANAGER LEGAL | Arry Wahyu |
| MANAGER PERSONALIA | Arry Wahyu |
| MANAGER AKUNTING | Feryandi |
| MANAGER KEUANGAN | Sugimin |
| MANAGER OPERASI | Capt Sukardi |
| MANAGER TEKNIK | Dradjat Soeprapto |
| MANAGER IT | I Wayan Parka |
| MANAGER MARKETING | Tomy Dwi Putra |

***Tabel*** *3.1 Struktur Organisasi Perusahaan*

### TIME SCHEDULE

Aktifitas penelitian ini secara bertahap keseluruhan dilaksanakan mulai bulan september 2017 sampai dengan bulan november 2017

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| NO | KEGIATAN | SEPTEMBER | | | | OKTOBER | | | | NOVEMBER | | | |
| I | II | III | IV | I | II | III | IV | I | II | III | IV |
| 1 | Pengumpulan dan Analisa Data |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Penulisan Laporan |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Asistensi BAB I |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Asistensi BAB II |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | Asistensi BAB III |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 | Asistensi BAB IV |  |  |  |  |  |  |  |  |  |  |  |  |
| 7 | Asistensi BAB V |  |  |  |  |  |  |  |  |  |  |  |  |
| 8 | Analisa Software |  |  |  |  |  |  |  |  |  |  |  |  |
| 9 | Rancangan Bangun Program |  |  |  |  |  |  |  |  |  |  |  |  |
| 10 | Uji Coba Program |  |  |  |  |  |  |  |  |  |  |  |  |
| 11 | Implementasi Program |  |  |  |  |  |  |  |  |  |  |  |  |
| 12 | Final Check Naskah & Persiapan Ujian Skripsi |  |  |  |  |  |  |  |  |  |  |  |  |
| 13 | Ujian Skripsi |  |  |  |  |  |  |  |  |  |  |  |  |
| 14 | Revisi dan Penyerahan Skripsi |  |  |  |  |  |  |  |  |  |  |  |  |

***Tabel*** *3.2 Time Schedule*

## METODE PENGUMPULAN DATA

### Observasi

Metode observasi adalah suatu cara pengambilan data dengan menggunakan mata tanpa menggunakan alat bantu standar lain. Observasi mengharuskan seorang analis menjadi pengamat orang dan aktifitas yang dijalankan demi mempelajari alur sistem.[[33]](#footnote-33) Disini objek penelitian adalah di kantor pusat PT.Kalstar Aviation bertempat di BSD Serpong.

Melalui tahapan observasi pengumpulan data dilakukan dengan melihat langsung proses dan kegiatan bisnis yang berjalan pada divisi controlling sertifikasi PT.Kalstar Aviation.

### Metode Wawancara

Wawancara atau *interview* adalah proses tanya jawab demi memperoleh keterangan untuk tujuan penelitian secara tatap muka antara pewawancara dengan si penjawab/responden baik secara langsung ataupun tidak langsung menggunakan alat yang dinamakan *interview guide. [[34]](#footnote-34)* Berikut adalah contoh form wawancara yang digunakan.

|  |  |  |
| --- | --- | --- |
| **PEDOMAN WAWANCARA**  Komponen Wawancara: *Input, Proses, Output*  Nara Sumber :   1. Staff Controlling 2. Manager Controlling 3. Staff Finance 4. Manager Finance | | |
| No | Komponen/Sub | Substansi Pertanyaan |
| 1 | Input Profil Awak Pesawat | 1. Ada berapa jumlah awak pesawat yang dimiliki saat ini? 2. Data apa saja yang dimiliki setiap satu awak pesawat? |
| 2 | Input Jenis Sertifikasi | 1. Sertifikasi apa saja yang harus dimiliki awak pesawat agar bisa menjalankan tugasnya? |
| 3 | Proses Input Profile | 1. Siapa yang memiliki izin mengolah data profile? |
| 4 | Proses Input Jenis Sertifikasi | 1. Siapa yang diperbolehkan untuk mengolah data sertifikasi? |
| 3 | Output informasi sertifikasi | 1. Informasi apa saja yang dibutuhkan sebagai untuk laporan akhir ? 2. Siapa saja yang boleh melihat informasi tersebut |

***Tabel*** *3.3 Pedoman Wawancara*

### Kajian Pustaka

Studi pustaka mengharuskan peneliti mengumpulkan, membaca serta mempelajari teori yang ada pada buku maupun karya ilmiah yang pernah dilakukan sebelumnya. Hal ini sangat diperlukan guna sebagai pengumpulan data, selain itu studi pustaka adalah sebagai sarana perolehan ide tentang masalah terbaru untuk dirumuskan dalam penelitian. Dalam hal ini dilakukan pengumpulan studi pustaka berupa buku-buku sistem informasi, buku-buku database, jurnal penelitian, karya tulis internet bahkan dari penulisan skripsi lulusan-lulusan sebelumnya.

#### Penelitian Sebelumnya

Sebagai bahan perbandingan dalam penelitian yang dilakukan, disini diambil beberapa contoh jurnal serta skripsi yang pernah dibuat oleh lulusan dari luar kampus STMIK Muhammadiyah Jakarta.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Judul Penelitian | Th | Peneliti | Pembahasan |
| 1 | Perancangan Sistem Informasi Monitoring Skripsi | 2017 | Mochamad Reza Ramadhan, Lukito Edi Nugroho, Selo Sulistyo | Perancangan Sistem Informasi untuk memonitor skripsi dengan menggunakan metode waterfall. |
| 2 | Rancang Bangun Sistem Informasi Monitoring Data Performansi Mitra Speedy Berbasis Web Pada PT.Telkom Regional Timur(Persero) |  | Amirudin Ilhamsyah, Titik Lusiani, Teguh Sutanto | Membahas tentang monitoring kinerja para mitra telkom speedy. Aplikasi diperuntukan hanya untuk level manager. |
| 3 | Perancangan Sistem Monitoring Pengerjaan Skripsi pada STMIK Stikom Bali Berbasis Web |  | Putu Ramayasa, Ida Bagus Ketut Suryana Arnawa | Membahas tentang monitoring skripsi yang terjadi akibat kendala-kendala yang ada dalam penulisan skripsi baik kendala dari sisi mahasiswa maupun dari sisi dosen pembimbing |
| 4 | Rancang Bangun Sistem Informasi Monitoring Penyewaan Gedung dan Infrastruktur Teknologi Informasi. | 2014 | Alviani Wahyuni Suyodni | Membahas tentang perancangan aplikasi untuk monitoring penyewaan gedung serta infrastruktur yang ada dalam gedung tersebut. (Studi kasus pada PT.Indosat,Tbk) |
| 5 | Pembuatan Aplikasi Sistem Informasi Monitoring Kegiatan Mahasiswa Berbasis Web dan Android Client. | 2015 | Erdi Risnandar | Penelitian Ini membahas tentang monitoring kegiatan mahasiswa. |

***Tabel*** *3.4 Lima (5) Penelitian Sebelumnya*

## METODOLOGI PENELITIAN

### Fountain Model

Metode yang digunakan dalam penelitian ini adalah metode SDLC*(Systems Development Life Cycle)* model *fountain*. Menurut Azhar Susanto(2004) “SDLC adalah salah satu metode pengembangan sistem informasi yang popular saat sistem informasi pertama kali dikembangkan”[[35]](#footnote-35). Metode SDLC model *fountain* ini dirasa sangat cocok untuk penelitian mengingat keterbatasan waktu yang ada. Ditakutkan selama penelitian ada salah satu step yang tidak bisa didahulukan. Sehingga bisa didahulukan step yang lain yang sudah siap untuk dilakukan penelitian dan melanjutkan step sebelumnya di tahapan selanjutnya.

Model *Fountain* Meliputi tahapan sebagai berikut:

#### Analisys

Menganalisa kebutuhan pengguna dari segi kemudahan penggunaan sistem yang akan dibangun dan kefisiensian hasil dari sistem.[[36]](#footnote-36)

#### Requerements Spesifications (User & Software)

Mengumpulkan informasi tentang kebutuhan pengguna dari system yang akan dibangun. apa saja yang di perlukan serta bagaimana menyesuaikan system agar pengguna yang awam pun dapat dengan mudah menggunakan system tersebut.

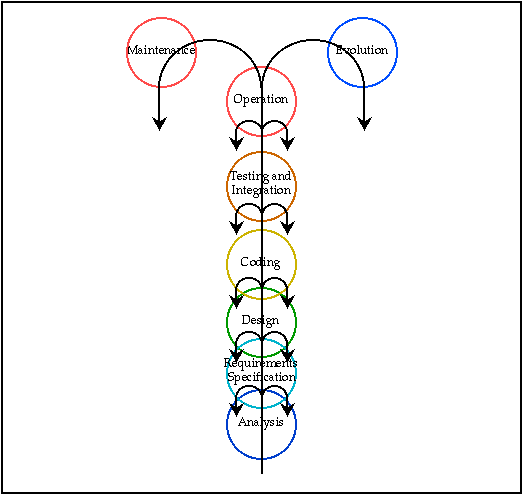
#### Design

Ada dua design yang dilakukan, pertama adalah design secara proses aplikasi tersebut berjalan kedua adalah design secara visual sederhana. Pada tahap design ini digunakan permodelan UML*(Unified Modelling Language)* sebagai standar baku internasional.[[37]](#footnote-37)

#### Coding

Coding adalah tahap dimana hasil dari design yang sudah dibuat sebelumnya dituangkan dalam bahasa yang dapat dikenal oleh komputer. Dalam hal ini digunakan bahasa pemrograman PHP yang menggunakan *Framework CodeIgniter* dengan menggunakan *database* MySQL.

#### Testing and Integration

Tahap testing dilakukan mulai dari submenu kemudian barulah *testing* sistem secara keseluruhan. Dalam tahap *testing* digunakan *black box testing* karena dalam tahap pengujian user pun nanti akan dilibatkan sehingga tidak memungkinkan user untuk bisa memahami coding yang dibuat, jadi cukup dengan *interfacce* yang ada.

#### Operation

Setelah *Testing* dirasa cukup, dan tidak ada kendala barulah sistem mulai dioperasikan kepada user.

#### Maintenance & Evolution

Sistem yang sudah jadi, perlu untuk dilakukan *maintenance*. Selain *maintenance* update sistem / *evolution* untuk menyesuaikan dengan perangkat-perangkat baru yang ada perlu juga dilakukan.

***Gambar 3.1*** *Fountain Model*

### Kelebihan dan kekurangan *Fountain* *Model*

#### Kelebihan:

Kelebihan dari metode SDLC Model *Fountain* adalah:

1. Memudahkan pengerjaan suatu penelitian karena step yang belum siap bisa dilewati dan bisa dikerjakan kembali pada step selanjutnya.
2. Kualitas dari sistem yang dihasilkan akan baik mengingat pengerjaanya bisa dilakukan secara bertahap.
3. Dokumen pengembangan sistem lebih terorganisir, karena walaupun dalam metode ini bisa melewati step sebelumnya, tidak diperbolehkan meninggalkan step yang sudah dikerjakan sebelum selesai pada step tersebut.

#### Kekurangan:

Sedangkan untuk kekuranganya antara lain:

1. Meskipun bisa melewati step yang sudah ditentukan, akan tetapi tidak diperbolehkan melewatkan antara step design dan coding.
2. Kesalahan kecil yang ditimbulkan kedepanya nanti bisa berkibat buruk jika tidak dibenahi dari awal.

# BAB IV

**HASIL DAN PEMBAHASAN**

## Analisa Sistem (*System Analysis*)

Dalam bab empat ini, dilakukan pembahasan yang mengacu pada identifikasi masalah yang ada. Permasalahan yang diangkat dalam penulisan skripsi ini adalah bagaimana membuat sistem informasi controlling sertifikasi awak pesawat yang bisa memudahkan staff controlling dalam mengontrol sertifikasi para aircrew.

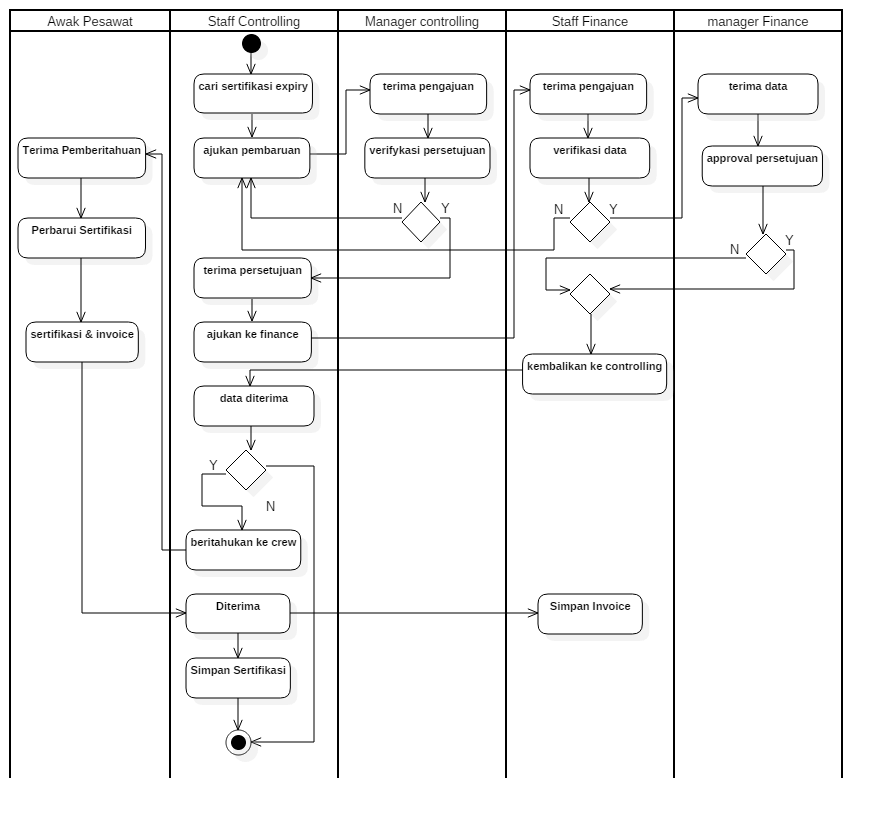
### Analisa Sistem Yang Berjalan

#### Prosedur sistem Berjalan

Berdasarkan analisa sistem yang dilakukan pada PT.Kalstar Aviation tepatnya pada bagian controlling sertifikasi, proses *controlling* sertifikasi awak pesawat saat ini masih menggunakan aplikasi pengolah angka yaitu *Microsoft Office Excel.* Controllingya pun masih menggunakan kolom-kolom yang ada pada aplikasi tersebut. Untuk lebih mengetahui mengenai prosedur sistem yang berjalan saat ini dijelaskan sebagai berikut menggunakan diagram aktifitas.

* Aktor Yang terlibat:

1. Staff Controlling : Bertugas menginputkan data *crew profile*, sertifikasi, mengontrol *Expiry*, mengajukan pembaruan sertifikasi dan menyimpan sertifikasi yang sudah diperbarui.
2. Manager Controlling: Bertugas mem*verify* pengajuan dari staff controlling serta sebagai penanggung jawab dari pengajuan sertifikasi yang akan diperbarui.
3. Staff Finance: bertugas sebagai *verify* data dari divisi finance untuk menghindari terjadinya dobel pengajuan, serta sebagai pihak yang menyimpan bukti perpanjangan sertifikasi yang sudah dilakukan pembaruan.
4. Manager Finance: Bertugas untuk menyetujui/menolak pengajuan pembaruan sertifikasi setelah melewati beberapa verifikasi.



***Gambar****. 4.1 Activity Diagram Sistem berjalan*

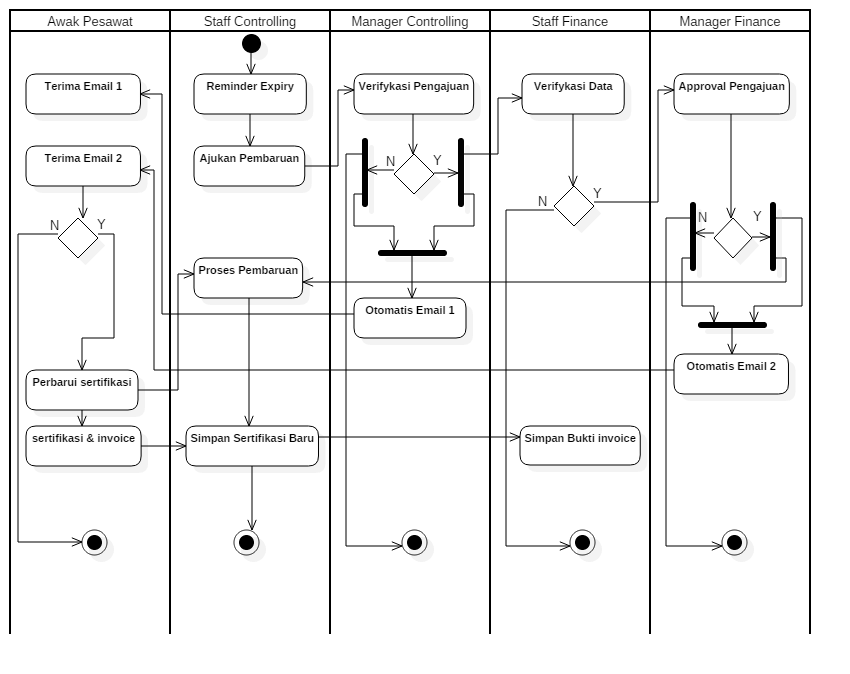
#### Penjelasan Sistem yang berjalan.

Berikut adalah penjelasan dari sistem yang berjalan pada perusahaan saat ini:

1. Staff Controlling mencari sertifikasi yang sudah *Expiry* pada data *excel* yang dimiliki.
2. Setelah data ditemukan, staff controling mengajukan pembaruan sertifikasi kepada manager controlling dalam bentuk 2 lembar *print out* yang nantinya 1 lembar di gunakan untuk pegangan *finance* dan 1 lembar digunakan untuk pegangan *staff controlling*.
3. Setelah didapatkan persetujuan dari *manager controlling*, selanjutnya diajukan ke pihak *finance* untuk dilakukan *verify*kasi.
4. Lolos *verify*kasi, staff finance langsung meneruskan pengajuan kepada manager finance.
5. Setelah disetujui oleh manager finance maka staff finance mengembalikan pengajuan yang sudah disetujui*(ditandangani)* oleh manager finance ke pihak controlling.
6. Staff controlling menerima persetujuan, dan memberitahukan ke awak pesawat untuk melakukan pembaruan sertifikasi.
7. Setelah awak pesawat melakukan pembaruan, sertifikasi terbaru dan bukti *invoice* diserahkan kembali ke pihak controlling dan pihak *finance*.

### Analisa Sistem yang diusulkan

Adapun analisa sistem yang diusulkan juga dituangkan dalam bentuk diagram aktifity, untuk diagramnya seperti berikut:



***Gambar****. 4.2 Activity Diagram Sistem usulan*

## Spesifikasi Kebutuhan *(Requirements Spesifications)*

### Kebutuhan Hadware

* Perangkat keras yang akan digunakan dalam pembuatan sistem:

1. Laptop Dell dengan processor Intel Core i3
2. RAM 4 GB
3. HDD 360 GB
4. Monitor, Keyboard, dan Mouse

* Perangkat keras yang digunakan dalam implementasi sistem:

1. Minimal Laptop/PC pentium 4
2. Minimal RAM 1GB
3. Minimal Hardisk 160GB
4. Monitor, keyboard, mouse dan printer

### Kebutuhan Software

* Perangkat lunak yang digunakan untuk pembuatan sistem:

1. Operating System windows 7
2. Sublime Text
3. WAMP Server
4. Browser Mozilla Firefox

* Perangkat lunak yang digunakan untuk implementasi:

1. Operating System Windows 7
2. WAMP Server
3. Browser Mozilla Firefox

### Kebutuhan Fungsional

Kebutuhan fungsional adala kebutuhan yang berisi proses-proses yang dilakukan oleh sistem juga berisi informasi-informasi yang harus ada dan dihasilkan oleh sistem.

1. Pendataan Profil awak pesawat.
2. User dapat menambahkan data awak pesawat yang aktif kembali, ataupun awak pesawat yang baru aktif bekerja.
3. User dapat menampilkan, mengubah mengedit dan menghapus data awak pesawat.
4. Controlling Sertifikasi *Expiry*.
5. User dapat dengan mudah melihat siapa-siapa awak pesawat yang sertifikasinya akan segera berakhir.
6. User dapat menggunakan data hasil tampilan sebagai acuan pengajuan pembaruan sertifikasi.
7. Progres pengajuan sertifikasi.
8. User dapat dengan mudah melakukan pengajuan sertifikasi tanpa harus menggunakan print out lagi.
9. User dapat melihat sejauh mana progres pengajuan sertifikasi yang diajukanya.
10. *Export* Laporan
11. Sistem dapat meng*Export* laporan hasil pengolahan data sertifikasi *Expiry*.
12. Sistem dapat meng*Export* list pengajuan pembaruan sertifikasi sesuai dengan tanggal yang ditentukan.

## Desain Sistem (System Design)

Untuk mendesain sistem disini digunakan UML diagram yang meliputi *use case diagram, class diagram, squence diagram* dan *aktifity diagram.*

### Use Case Digram Sistem



***Gambar****. 4.3 Use Case Diagram Integrasi Digital*

##### Deskripsi Use Case (Lihat *certifi Expiry*)

|  |  |
| --- | --- |
| **Aktor Utama:**  Staff Controlling  **Aktor Tambahan:**  Staff Finance,  Manager Controlling,  Manager Finance  **Nama Use Case:**  Lihat Sertifikasi *Expiry*  **Tujuan Use Case:**  Menampilkan data sertifikasi keseluruhan awak pesawat baik yang sudah *Expiry* maupun yang masih berlaku.  **Kondisi Awal:**  Sistem sudah terbuka, berada pada halaman Utama Home, memiliki izin akses ‘Y’ pada modul tcontrolling*Expiry* field ‘lihat’.  **Kondisi Akhir:**  Tampil tabel controlling *Expiry* sesuai jenis crew.  **Trigger:**  Pilih menu information | **Optimistic Flow:**   1. User Masuk ke Sistem Controlling Sertifikasi Awak Pesawat. 2. User memilih menu information. 3. User memilih sub menu view controlling. 4. User menselect jenis awak pesawat. 5. User mengklik tombol lihat data. 6. Tabel Controlling Sertifikasi menampilkan data awak pesawat berdasarkan jenis.   **Pesimistic Flow:**   1. 4.a Hasil tidak ditemukan – Tabel menampilkan pesan “data not available” |

##### Deskripsi Use (Case Lihat *List Certifi*)

|  |  |
| --- | --- |
| **Aktor Utama:**  Staff Controlling  **Aktor Tambahan:**  Staff Finance,  Manager Controlling,  Manager Finance  **Nama Use Case:**  Lihat List Sertifikasi(per profil)  **Tujuan Use Case:**  Menampilkan data sertifikasi yang dimiliki per awak pesawat.  **Kondisi Awal:**  Sistem sudah menampilkan tabel sertifikasi *Expiry*, memiliki izin akses ‘Y’ pada modul tcontrolling*certifi* field ‘lihat’.  **Kondisi Akhir:**  Tampil tabel list sertifikasi sesuai nama crew.  **Trigger:**  Pilih tombol lihat detail list | **Optimistic Flow:**   1. User sudah berada pada halaman lihat sertifikasi *Expiry*. 2. User memilih detail list sertifikasi. 3. Tabel List Sertifikasi menampilkan data sertifikasi yang dimiliki oleh awak pesawat tersebut.   **Pesimistic Flow:**   1. 3.a Hasil tidak ditemukan – Tabel menampilkan pesan “data not available” |

##### Deskripsi Use Case (tambah *list certifi*)

|  |  |
| --- | --- |
| **Aktor Utama:**  Staff Controlling  **Aktor Tambahan:**  -  **Nama Use Case:**  Tambah List Sertifikasi(per profil)  **Tujuan Use Case:**  Menambah data sertifikasi yang dimiliki awak pesawat.  **Kondisi Awal:**  Sistem sudah menampilkan tabel list sertifikasi per awak pesawat, memiliki izin akses ‘Y’ pada modul tcontrolling*certifi* field ‘tambah’.  **Kondisi Akhir:**  Sistem menampilkan pesan “tambah list sertifikasi berhasil”  **Trigger:**  Pilih tombol tambah | **Optimistic Flow:**   1. User sudah berada pada halaman lihat list sertifikasi. 2. User mengklik tombol tambah. 3. Form tambah list sertifikasi ditampilkan. 4. User menselect sertifikasi yang akan ditambahkan. 5. User mengisi tanggal issued dan tanggal *Expiry* sertifikasi 6. User mengklik tombol proses tambah 7. Sistem menampilkan “tambah list sertifikasi berhasil”   **Pesimistic Flow:**   1. 8.a Data sudah ada – sistem menampilkan pesan “data sudah ada” |

##### Deskripsi Use Case (Edit *List Certifi*)

|  |  |
| --- | --- |
| **Aktor Utama:**  Staff Controlling  **Aktor Tambahan:**  Staff Finance,  Manager Controlling,  Manager Finance  **Nama Use Case:**  Edit List Sertifikasi(per profil)  **Tujuan Use Case:**  Mengedit data sertifikasi yang dimiliki awak pesawat.  **Kondisi Awal:**  Sistem sudah menampilkan tabel list sertifikasi per awak pesawat, memiliki izin akses ‘Y’ pada modul tcontrolling*certifi* field ‘edit’.  **Kondisi Akhir:**  Sistem menampilkan pesan “edit list sertifikasi berhasil”  **Trigger:**  Pilih tombol edit | **Optimistic Flow:**   1. User sudah berada pada halaman lihat list sertifikasi. 2. User mengklik tombol edit. 3. Form edit list sertifikasi ditampilkan. 4. User mengganti data. 5. User mengklik tombol proses edit. 6. Sistem menampilkan “edit list sertifikasi berhasil”   **Pesimistic Flow:**   1. 6.a Data digunakan – sistem menampilkan pesan “data sedang digunakan” |

##### Deskripsi Use Case (ajukan *renewal*)

|  |  |
| --- | --- |
| **Aktor Utama:**  Staff Controlling  **Aktor Tambahan:**  -  **Nama Use Case:**  Ajukan Pembaruan(sertifikasi)  **Tujuan Use Case:**  Mengajukan pembaruan sertifikasi *Expiry* yang dimiliki oleh awak pesawat.  **Kondisi Awal:**  Sistem sudah menampilkan tabel list sertifikasi per awak pesawat, memiliki izin akses ‘Y’ pada modul t*certifirenewal* akses field ‘lihat’.  **Kondisi Akhir:**  Sistem menampilkan pesan “pengajuan *renewal* berhasil”  **Trigger:**  Klik link tanggal pada sertifikasi yang akan di ajukan. | **Optimistic Flow:**   1. User sudah berada pada halaman lihat list sertifikasi. 2. User mengklik link aktif pada tanggal yang *Expiry*. 3. Form pengajuan pembaruan list sertifikasi ditampilkan. 4. User mengisi Form pengajuan. 5. User mengklik tombol proses ajukan. 6. Sistem menampilkan “pengajuan pembaruan list sertifikasi berhasil”   **Pesimistic Flow:**   1. Data sedang diajukan/belum waktunya diajukan – link pengajuan pembaruan sertifikasi pada tanggal tidak aktif. |

##### Deskripsi Use Case (lihat *list renewal*)

|  |  |
| --- | --- |
| **Aktor Utama:**  Staff Controlling  Staff Finance,  Manager Controlling,  Manager Finance  **Aktor Tambahan:**  -  **Nama Use Case:**  Lihat list pengajuan(pembaruan)  **Tujuan Use Case:**  Menyajikan informasi nama-nama yang diajukan pembaruan sertifikasinya.  **Kondisi Awal:**  Sistem sudah terbuka, berada pada halaman Utama Home, memiliki izin akses ‘Y’ pada modul t*certifirenewal* field ‘lihat’.  **Kondisi Akhir:**  Sistem menampilkan nama-nama yang diajukan pembaruan sertifikasinya  **Trigger:**  Pilih menu informasi | **Optimistic Flow:**   1. User Masuk ke Sistem Controlling Sertifikasi Awak Pesawat. 2. User memilih menu information. 3. User memilih sub menu view *renewal*. 4. Tabel *Certifi* renewl menampilkan data sertifikasi yang diajukan.   **Pesimistic Flow:**   1. 3.a Hasil tidak ditemukan – Tabel menampilkan pesan “data not available” |

##### Deskripsi Use Case (detail *renewal*)

|  |  |
| --- | --- |
| **Aktor Utama:**  Staff Controlling  Staff Finance  Manager Controlling,  Manager Finance  **Aktor Tambahan:**  -  **Nama Use Case:**  Detail Pengajuan(pembaruan)  **Tujuan Use Case:**  Melihat informasi rinci tentang detail sertifikasi yang di ajukan.  **Kondisi Awal:**  Sistem sudah terbuka, dan suda menampilkan tabel tabel list pengajuan.  **Kondisi Akhir:**  Sistem menampilkan detail dari sertifikasi yang diajukan.  **Trigger:**  Klik tombol detail | **Optimistic Flow:**   1. User sudah masuk ke Sistem dan berada pada pada halaman lihat list pengajuan. 2. User mengklik tombol detail. 3. Sistem menampilkan detail data yang diminta.   **Pesimistic Flow:**   1. 3.a User tidak memiliki ijin – tombol detail tidak tampil. |

##### Deskripsi Use Case (*verify* 1)

|  |  |
| --- | --- |
| **Aktor Utama:**  Manager Control  **Aktor Tambahan:**  -  **Nama Use Case:**  Acc mgr(manager controlling)  **Tujuan Use Case:**  Mem*verify* pengajuan pembaruan sertifikasi dari staff controlling.  **Kondisi Awal:**  Sistem sudah terbuka, dan suda menampilkan tabel list pengajuan.  **Kondisi Akhir:**  Sistem menampilkan pesan berhasil di *verify*  **Trigger:**  Klik tombol *verify* | **Optimistic Flow:**   1. User sudah masuk ke Sistem dan berada pada pada halaman lihat list pengajuan. 2. User mengklik tombol *verify*. 3. Sistem menampilkan form *verify*. 4. User menselect *verify* 5. User mengisi value 6. User mengklik tombol *verify* 7. Sistem menampilkan pesan “berhasil di verifikasi”   **Pesimistic Flow:**   1. 7.a Gagal di verifikasi – sistem menampilkan pesan “data sudah di verifikasi”. |

##### Deskripsi Use Case (*verify* 2)

|  |  |
| --- | --- |
| **Aktor Utama:**  Staff Finance  **Aktor Tambahan:**  -  **Nama Use Case:**  *Verify*(staff finance)  **Tujuan Use Case:**  Mem*verify* pengajuan pembaruan sertifikasi yang sudah di acc manager controlling.  **Kondisi Awal:**  Sistem sudah terbuka, dan suda menampilkan tabel list pengajuan.  **Kondisi Akhir:**  Sistem menampilkan pesan berhasil di *verify*  **Trigger:**  Klik tombol *verify* | **Optimistic Flow:**   1. User sudah masuk ke Sistem dan berada pada pada halaman lihat list pengajuan. 2. User mengklik tombol *verify*. 3. Sistem menampilkan form *verify*. 4. User menselect *verify* 5. User mengisi value 6. User mengklik tombol *verify* 7. Sistem menampilkan pesan “berhasil di verifikasi”   **Pesimistic Flow:**   1. 7.a Gagal di verifikasi – sistem menampilkan pesan “data sudah di verifikasi”. |

##### Deskripsi Use Case (*Approve*)

|  |  |
| --- | --- |
| **Aktor Utama:**  Manager Finance  **Aktor Tambahan:**  -  **Nama Use Case:**  *Approve*(manager finance)  **Tujuan Use Case:**  Menyeteujui pengajuan pembaruan sertifikasi yang sudah di di ajukan oleh staff controlling dan melalui dua verfikasi.  **Kondisi Awal:**  Sistem sudah terbuka, dan suda menampilkan tabel list pengajuan.  **Kondisi Akhir:**  Sistem menampilkan pesan berhasil di *Approve*  **Trigger:**  Klik tombol *Approve* | **Optimistic Flow:**   1. User sudah masuk ke Sistem dan berada pada halaman lihat list pengajuan. 2. User mengklik tombol *Approve*. 3. Sistem menampilkan form *Approve*. 4. User menselect *Approve* 5. User mengisi value 6. User mengklik tombol *Approve* 7. Sistem menampilkan pesan “berhasil di *Approve*”   **Pesimistic Flow:**   1. 7.a Gagal di *Approve* – sistem menampilkan pesan “data sudah di *Approve*”. |

##### Deskripsi Use Case (*save* new *certifi* & *save* invoice)

|  |  |
| --- | --- |
| **Aktor Utama:**  Staff Controlling  Staff Finance  **Aktor Tambahan:**  -  **Nama Use Case:**  *Save* sertifikasi & *save* bukti.  **Tujuan Use Case:**   * Untuk staf controlling: menyimpan sertifikasi yang sudah di perbarui. * Untuk staff finance: menyimpan bukti invoice pembaruan sertifikasi.   **Kondisi Awal:**  Sistem sudah terbuka, dan suda menampilkan tabel list pengajuan.  **Kondisi Akhir:**  Sistem menampilkan pesan berhasil di *save*  **Trigger:**  Klik tombol *save* | **Optimistic Flow:**   1. User sudah masuk ke Sistem dan berada pada halaman lihat list pengajuan. 2. User mengklik tombol *save*. 3. Sistem menampilkan form. 4. User mengisi value 5. User mengklik tombol *save* 6. Sistem menampilkan pesan “berhasil di *save*”   **Pesimistic Flow:**   1. 6.a Gagal di simpan – sistem menampilkan pesan “data sudah di simpan sebelumnya”. |

##### Deskripsi Use Case(*Export* list *renewal*)

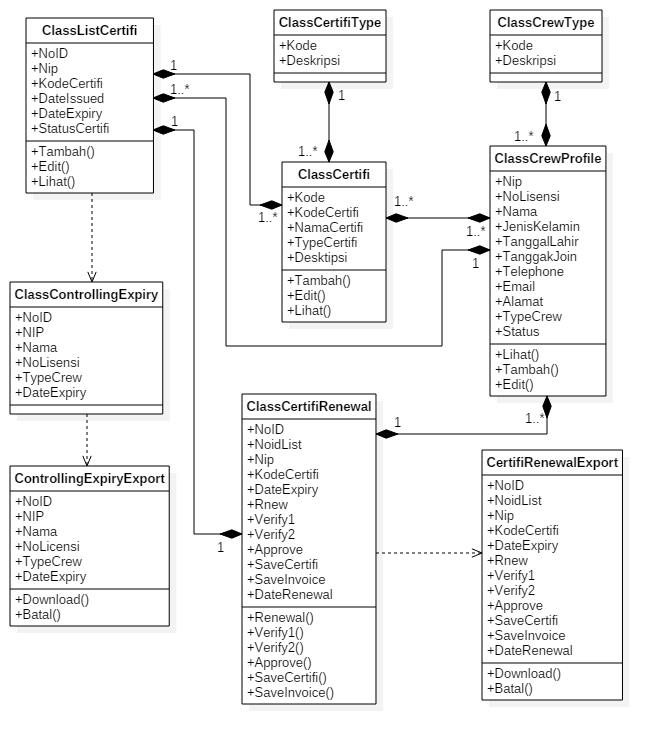
|  |  |
| --- | --- |
| **Aktor Utama:**  Staff Finance  **Aktor Tambahan:**  Staff Controlling  **Nama Use Case:**  *Export* List Pengajuan  **Tujuan Use Case:**  Memberikan laporan hasil akhir pengelolaan  **Kondisi Awal:**  Sistem sudah terbuka, berada pada halaman Utama Home, memiliki izin akses ‘Y’ pada modul *Export* field ‘lihat’.  **Kondisi Akhir:**  Laporan berhasil di *Export*  **Trigger:**  Pilih menu *Export* | **Optimistic Flow:**   1. User memilih menu *Export*. 2. Sistem menampilkan form range tanggal. 3. User mengisi Range tanggal. 4. User mengklik *Export* data. 5. Sistem menampilkan pop up download. 6. User mengklik tombol *save*. 7. Laporan berhasil di *Export*   **Pesimistic Flow:**   1. 7.a Gagal download |

##### Deskripsi Use Case(*Export* *Certifi* *Expiry*)

|  |  |
| --- | --- |
| **Aktor Utama:**  Staff Controlling  **Aktor Tambahan:**  Staff Finance  **Nama Use Case:**  *Export* Sertifikasi *Expiry*  **Tujuan Use Case:**  Memberikan laporan hasil akhir pengelolaan  **Kondisi Awal:**  Sistem sudah terbuka, berada pada halaman Utama Home, memiliki izin akses ‘Y’ pada modul *Export* field ‘lihat’.  **Kondisi Akhir:**  Laporan berhasil di *Export*  **Trigger:**  Pilih menu *Export* | **Optimistic Flow:**   1. User memilih menu *Export*. 2. Sistem menampilkan form select dropdown jenis awak pesawat. 3. User menselect jenis awak pesawat. 4. User mengklik tombol *Export*. 5. Sistem menampilkan pop up download. 6. User mengklik tombol *save*. 7. Laporan berhasil di *Export*   **Pesimistic Flow:**   1. 7.a Gagal download |

### Class Diagram

Setelah memodelkan sistem utama dengan use case diagram, selanjutnya disini di analisa class-class yang dapat di simpulkan dalam use case diagram, class-class tersebut adalah class yang memiliki hubungan khusus dengan use case diagram yang dibuat.

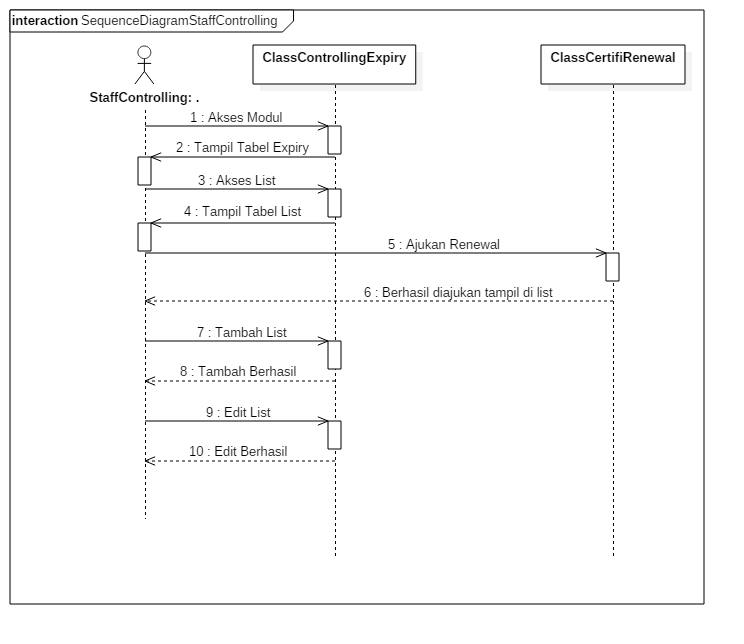


***Gambar****. 4.4 Class Diagram*

### Squence Diagram

Squence diagram adalah diagram yang digunakan untuk menjelaskan lebih rinci dari usecase diagram yang sebelumnya dibentuk.

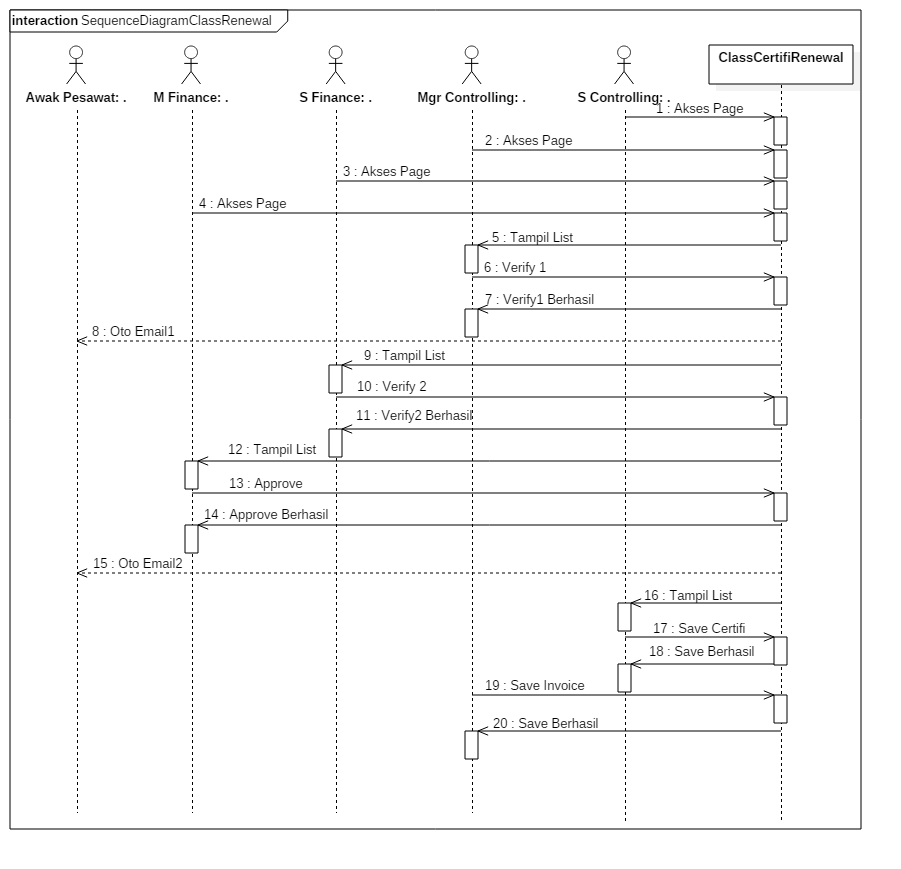
#### Sequence Staff Controlling



***Gambar****. 4.5 Sequence diagram staff controlling*

Sequence Staff Controlling menjelaskan alur kerja apa saja yang dilakukan oleh aktor staff controlling dalam aplikasi ini. Dimulai dari akses modul kemudian aplikasi akan menampilkan data pada tabel *Expiry*. Selanjutnya aktor mengakses list, sistem menampilkan tabel list sertifikasi. Operasi yang dapat dilakukan adalah menambah list sertifikasi, mengedit list sertifikasi dan mengajukan pembaruan sertifikasi.

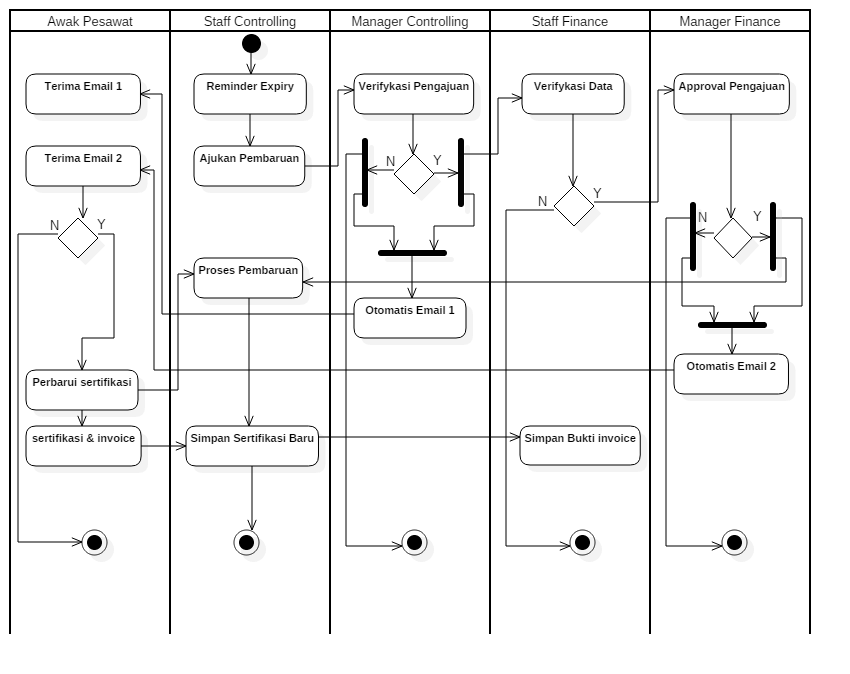
#### Sequence Diagram Class Certifi Renewal



***Gambar****. 4.6 Sequence diagram class certifi renewal*

Sequence diagram class *certifi* *renewal* disini menjelaskan alur proses class *certifi* *renewal* yang saling berkesinambungan. Dimulai dari *verify* 1 oleh manager controlling, selanjutnya *verify* 2 oleh staff finance dan setelah dipastikan semua *verify* benar maka dilanjutkan approval oleh manager finance. Setalah proses pembaruan sertifikasi selesai, aktor staff controlling menyimpan sertifikasi yang baru dan aktor staff finance menyimpan bukti invoice.

### Activity Diagram



***Gambar****. 4.7 Aktifity diagram sistem informasi controlling sertifikasi.*

Pada diagram actifity, disini dijelaskan aktifitas mulai dari reminder sertifikasi *Expiry* sampai dengan penyimpanan sertifikasi baru dan bukti invoice. Proses tambah, edit, input serta aktifitas umum lainya tidak dijelaskan, karena dirasa hal tersebut sama dengan proses-proses pada aplikasi yang lain.

Penjelasan diagram aktifitas diatas sebagai berikut:

* Aktifitas dimulai dari staff controlling yang melihat reminder bahwa ada sertifikasi yang harus segera diperbarui.
* Maka diambillah langkah pengajuan pembaruan sertifikasi.
* Setelah diajukan, manager controlling mem*verify*kasi pengajuan apakah layak atau tidak untuk diajukan, jika tidak “N” maka aktifitas selesai, jika iya “Y” lanjutkan ke *verify*kasi staff finance. Saat manager controlling mengklik tombol submit baik itu berupa penolakan pengajuan ataupun persetujuan, maka sistem otomatis akan mengirimkan pemberitahuan via email kepada awak pesawat yang terkait pengajuanya sesuai dengan keputusan manager.
* Staff Finance mem*verify*kasi data pengajuan apakah terdapat data yang sama penah diajukan tidak sesuai dengan jadwal, atau ada hal-hal lain yang perlu untu di filter. Jika *verify*kasi tidak “N” maka aktifitas selesai, jika *verify*kasi iya “Y” maka lanjutkan ke approval dari manager finance.
* Manager Finance memutuskan sesuai dengan laporan finance sebelumnya serta melihat dari kondisi keuangan yang ada. Jika approval dari manager finance tidak”N” maka aktifitas selesai, jika iya”Y” maka kembalikan ke staff controlling untuk dilakukan pemrosesan pembaruan. Baik kondisi yang di hasilkan N atau Y otomatis sistem akan mengirimkan pemberitahuan ke awak pesawat. Jika pengajuan disetujui, maka awak pesawat segera melakukan pembaruan sertifikai.
* Staff Controlling menunggu proses pembaruan dari awak pesawat.
* Setelah pembaruan selesai, awak pesawat menyerahkan sertifikasi yang sudah diperbarui ke staff controlling dan bukti pembaruan kepada pihak finance.
* Staff controlling menyimpan sertifikasi yang baru, staff finance menyimpan bukti pembaruam.
* Aktifitas selesai.

### Design Database

Desain database sangat diperlukan mengingat aplikasi controlling yang dibuat ini melibatkan database sebagai penampung data. Baik data yang belum diolah sampai dengan data yang sudah diolah tertampung dalam data base.

#### Tabel user account (tpengguna)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO | Field | *Type* | Deskripsi Field | Ket |
| 1  2  3  4  5  6 | no\_id  username  password  nama  level  foto | *Varchar(10)*  *Varchar(30)*  *Varchar(40)*  *Varchar(30)*  *Varchar(6)*  *Varchar(20)* | Nomor ID  Nama User  Password User  Nama  Level  Foto Upload | *Key* |

***Tabel*** *4.1 tpengguna*

Tabel user account (tpengguna) berfungsi untuk menyimpan data user yang di perbolehkan untuk mengakses sistem. Sehingga apabila user belum terdaftar dalam tabel tersebut tidak bisa mengakses aplikasi.

#### Tabel modul akses(tmodulakses)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO | Field | *Type* | Deskripsi Field | Ket |
| 1  2  3  4  5  6  7 | no  id\_user  nama\_modul  lihat  tambah  edit  delete | Int(11)  Varchar(10)  Varchar(15)  Varchar(1)  Varchar(1)  Varchar(1)  Varchar(1) | Nomor  ID user  Nama modul  Lihat  Tambah  Edit  Delete | *Key (ai)* |

***Tabel*** *4.2 tmodulakses*

Tabel modul akses(tmodulakses) berfungsi sebagai pengatur hak akses user boleh tidaknya mengakses sebuah modul yang disediakan. Setiap modul memiliki 4 hak akses seperti lihat, tambah, edit dan delete.

#### Tabel Type Crew (tcrewType)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO | Field | *Type* | Deskripsi Field | Ket |
| 1  2  3 | noid  code  description | Int(11)  Varchar(6)  Varchar(25) | Nomor ID  Kode  Kepanjangan | *Key (ai)* |

***Tabel*** *4.3 tcrewtype*

Tabel *Type* Crew*(tcrewType)* berguna sebagai data master *Type* crew yang nantinya akan tampil sebagai select dropdown.

#### Tabel Type Crew Certifi(tcrewcertifiType)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO | Field | *Type* | Deskripsi Field | Ket |
| 1  2  3 | noid  code  description | Int(11)  Varchar(3)  Varchar(25) | Nomor ID  Kode  Kepanjangan | Key *(ai)* |

***Tabel*** *4.4 tcrewcertifitype*

Tabel *Type* Crew *Certifi(tcrewcertifiType)* berguna sebagai data master *Type* crew *certifi*yang nantinya akan tampil sebagai select dropdown.

#### Tabel Certifi Type(tcertifiType)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO | Field | *Type* | Deskripsi Field | Ket |
| 1  2  3  4  5  6  7 | noid  code\_auto  code\_cert  name\_cert  for\_crew  *certifi*\_*Type*  description\_cert | Int(11)  Varchar(6)  Varchar(8)  Varchar(25)  Varchar(2)  Varchar(3)  text | Nomor ID  Kode Otomatis  Kode Sertifikasi  Nama Sertifikasi  Untuk Crew  *Type* Sertifikasi  Deskripsi Sertifikasi | Key *(ai)* |

***Tabel*** *4.5 tcertifitype*

Tabel *Type* *Certifi(tcertifiType)* berguna sebagai data master *Type* *certifi* yang nantinya akan tampil sebagai select dropdown sebagai pilihan saat melakukan insert ataupun edit tabel information.

#### Tabel Crew Profile(tcrewprofile)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO | Field | *Type* | Deskripsi Field | Ket |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14 | noid  nip  license\_no  first\_name  last\_name  gender  date\_of\_birth  date\_of\_join  phone1  phone2  email  address  passport  crew\_*Type* | *Int(11)*  *Varchar(12)*  *Varchar(10)*  *Varchar(15)*  *Varchar(15)*  *Varchar(2)*  *Date*  *Date*  *Varchar(13)*  *Varchar(13)*  *Varchar(30)*  *Text*  *Varchar(12)*  *Varchar(2)* | Nomor ID  No Induk Pegawai  No Lisense  Nama Depan  Nama Akhir  Jenis Kelamin  Tanggal Lahir  Tanggal Bergabung  No Telefon 1  No Telefon 2  Alamat Email  Alamat  No Passport  Tipe awak pesawat | *Key (ai)* |

***Tabel*** *4.6 tcrewprofile*

Tabel Crew Profile*(tcrewprofile)* berguna sebagai data master crew profile yang nantinya akan tampil sebagai select dropdown sebagai pilihan saat melakukan insert ataupun edit tabel information.

#### Tabel Controlling Sertifikasi (tcontrollingcertifi)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO | Field | *Type* | Deskripsi Field | Ket |
| 1  2  3  4  5  6  7  8  9  10 | Noid  nip  code*certifi*\_auto  date\_issued  date\_*Expiry*  cert\_status  create\_by  create\_date  update\_by  update\_date | *Int(11)*  *Varchar(12)*  *Varchar(6)*  *Date*  *Date*  *Varchar(4)*  *Text*  *Text*  *Text*  *Text* | Nomor ID  No Induk Pegawai  Kode Sertifikasi  Tanggal diaktifkan  Tanggal berakhir  Status sertifikasi  Dibuat oleh  Tanggal dibuat  Diupdate oleh  Tanggal diupdate | Key *(ai)* |

***Tabel*** *4.7 tcontrollingcertifi*

Tabel Controling sertifikasi *(tcontrollingcertifi)* berfungsi sebagai tabel yang nantinya akan digunakan sebagai acuan controlling oleh staff yang melakukan tugasnya.

#### Tabel Controlling Expiry(tcontrollingExpiry)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO | Field | *Type* | Deskripsi Field | Ket |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23 | Noid  nip  first\_name  license\_no  crew\_*Type*  status\_crew  cr001  cr002  cr003  cr004  cr005  cr006  cr007  cr008  cr009  cr010  cr011  cr012  cr013  cr014  cr015  cr016  cr017 | *Int(11)*  *Varchar(12)*  *Varchar(15)*  *Varchar(10)*  *Varchar(2)*  *Varchar(2)*  *Date*  *Date*  *Date*  *Date*  *Date*  *Date*  *Date*  *Date*  *Date*  *Date*  *Date*  *Date*  *Date*  *Date*  *Date*  *Date*  *date* | Nomor ID  No Induk Pegawai  Nama Depan  No Lisense  *Type* Awak Pesawat  Status Aktif/Tidak  Nama field sesuai ID  =  =  =  =  =  =  =  =  =  =  =  =  =  =  =  = | Key *(ai)* |

***Tabel*** *4.8 tcontrollingexpiry*

Tabel tcontrolligExpiry, adalah tabel yang berfungsi untuk menyimpan dan menampilkan data sertifikasi yang Expiry yang selanjutnya akan dilakukan aksi pengajuan pembaruan oleh staff controlling.

#### Tabel Certifi Renewal (tcertifirenewal)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| NO | Field | *Type* | Deskripsi Field | Ket |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22 | no  noid\_list  nip  *certifi*\_code  date\_*Expiry*  request\_renew  date\_rr  remark\_rr  request\_*verify*\_mc  date\_rvmc  remark\_rvmc  request\_*verify*\_sf  date\_rvsf  remark\_rvsf  approval\_request\_mf  date\_armf  remark\_armf  box\_*save*\_noted  *save*\_noted  box\_*save*\_*certifi*  *save*\_*certifi*  date\_*renewal* | *Int(11)*  *Int(11)*  *Varchar(12)*  *Varchar(8)*  *Date*  *Varchar(4)*  *Date*  *Text*  *Varchar(4)*  *Date*  *Text*  *Varchar(4)*  *Date*  *Text*  *Varchar(4)*  *Date*  *Text*  *Text*  *Varchar(2)*  *Text*  *Varchar(2)*  *Date* | Nomor ID  ID List  No Induk Pgw  Kode Sertfikasi  Berakhir  Diajukan  Tgl Diajukan  Catatan  Verifikasi1  Tgl Verifikasi1  Catatan  Verifikasi2  Tgl Verifikasi2  Catatan  Disetujui  Tgl Disetujui  Catatan  Catatan Simpan  Simpan  Catatan Simpan  Simpan  Tgl Diperbarui | Key *(ai)* |

***Tabel*** *4.9 tcertifirenewal*

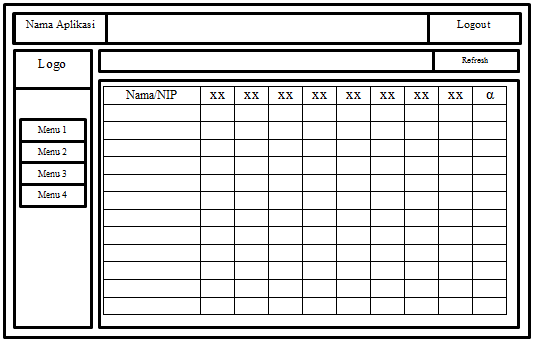
Tabel t*certifirenewal* adalah tabel yang digunakan untuk menampung data-data pengajuan yang diajukan. Tabel tersebut sangat erat hubunganya dengan tabel tcontrolling*Expiry* dan tcontrollingcertfi.

### Rancangan Interface

Rancangan interface disini dibuat segai rancangan bagaimana tampilan jadi dari aplikasi yang akan dibuat baik dari sisi input maupun output. Rancangan tersebut masih bersifat global, sehingga kemungkinan nanti berbeda dengan aplikasi saat sudah jadi, akan tetapi tidak terlalu signifikan perbedaanya.

#### Rancangan Interface Output

##### Output view controlling *Expiry*.



***Gambar****. 4.8 desain output view controlling expiry.*

Keterangan:

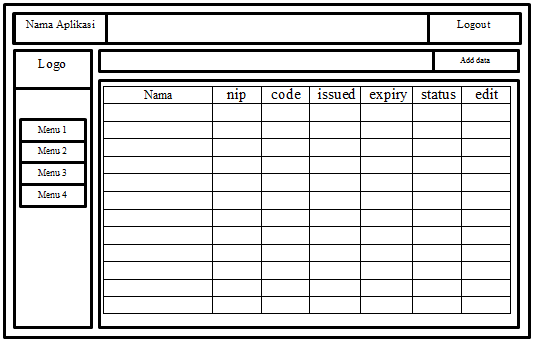
Menu 1,2,3 dan 4 : Main Menu.

Nama/Nip : Nama dan nip awak pesawat

xx : sertifikasi yang dimiliki awak pesawat berikut waktu *Expiry*nya

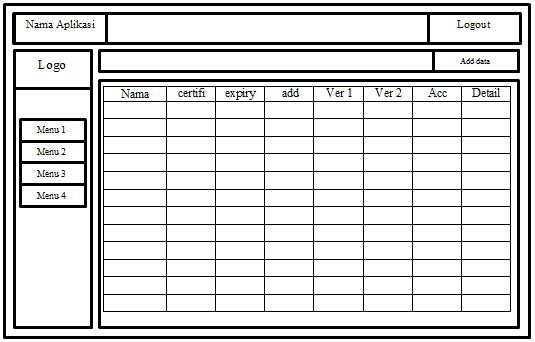
α : view detail list

##### Output view list controlling *certifi*



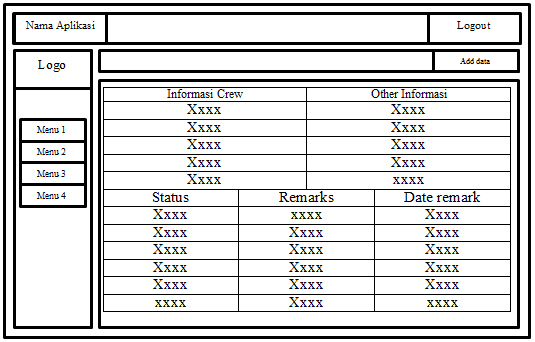
***Gambar****. 4.9 desain output view list controlling certifi.*

##### Output view *certifi* *renewal*



***Gambar****. 4.10 desain output view certifi renewal.*

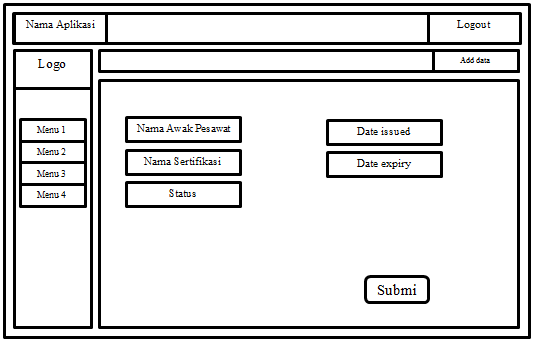
##### Output view detail *certifi* *renewal*



***Gambar****. 4.11 desain output view detail certifi renewal.*

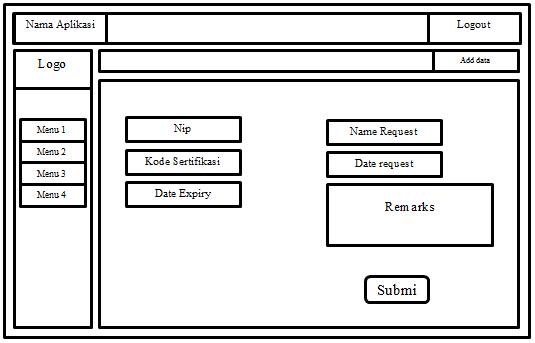
#### Rancangan Interface Input

##### Input view add / edit *certifi* list



***Gambar****. 4.12 desain input view add/edit certifi list.*

##### Input View Request *Renewal*, *Verify* dan Approval



***Gambar****. 4.13 desain input view request renewal, verify dan approval.*

## Testing Aplikasi (Application Testing)

Testing aplikasi atau yang lebih dikenal dengan pengujian sistem sangat diperlukan untuk menjamin kualitas dari aplikasi yang dibuat. Selain itu testing aplikasi bertujuan untuk menemukan bug sistem dari awal sehingga saat sistem sudah diimplementasikan meskipun terjadi bug hanya dalam presentasi yang kecil. Metode pengujuan sistem disini menggunakan metode pengujian black box. Pengujian black box ini tidak perlu mengetahui sampai seluk beluk pengcodingan yang terjadi di belakang layar, cukup mengetahui bagaimana kesesuaian hasil output atas input yang di lakukan.

### Rencana Pengujian

Pengujian sistem informasi controlling sertifikasi awak pesawat menggunakan data uji berupa data masukan dari user[[38]](#footnote-38).

#### Pengujian Login / Log out

|  |  |  |  |
| --- | --- | --- | --- |
| No | Sub Modul yang diuji | Detail Uji | Jenis Pengujian |
| 1 | Login | User melakukan login | *Black Box* |
| 2 | Logout | User melakukan logout | *Black Box* |

**Tabel** *4.10 Pengujian Login*

#### Pengujian Group Account

|  |  |  |  |
| --- | --- | --- | --- |
| No | Sub Modul yang diuji | Detail Uji | Jenis Pengujian |
| 1 | Tambah User | Melakukan penambahan data user | *Black Box* |
| 2 | Edit User | Melakukan pengeditan data user | *Black Box* |
| 3 | Edit Password | Melakukan Penditan Password User | *Black Box* |
| 4 | Edit Hak Akses | Melakukan Edit Hak Akses User | *Black Box* |
| 5 | Hapus User | Menghapus User | *Black Box* |

**Tabel** *4.11 Pengujian Group Account*

#### Pengujian Group Data Support

|  |  |  |  |
| --- | --- | --- | --- |
| No | Sub Modul yang diuji | Detail Uji | Jenis Pengujian |
| 1 | Tambah Crew *Type* | User melakukan tambah crew *Type* | *Black Box* |
| 2 | Tambah Crew *Certifi* | User melakukan tambah crew *certifi* | *Black Box* |
| 3 | Tambah *Certifi* *Type* | User melakukan tambah *certifi* *Type* | *Black Box* |
| 4 | Tambah Crew Profile | User melakukan tambah crew profile | *Black Box* |
| 5 | Edit Crew *Type* | User melakukan edit crew *Type* | *Black Box* |
| 6 | Edit Crew *Certifi* | User melakukan edit crew *certifi* | *Black Box* |
| 7 | Edit *Certifi* *Type* | User melakukan edit *certifi* *Type* | *Black Box* |
| 8 | Edit Crew Profile | User melakukan edit crew profile | *Black Box* |
| 9 | Hapus Crew *Type* | User melakukan hapus crew *Type* | *Black Box* |
| 10 | Hapus Crew *Certifi* | User melakukan hapus crew *certifi* | *Black Box* |
| 11 | Hapus *Certifi* *Type* | User melakukan hapus *certifi* *Type* | *Black Box* |

**Tabel** *4.12* *Pengujian Group Data Suport*

#### Pengujian Group Information

|  |  |  |  |
| --- | --- | --- | --- |
| No | Sub Modul yang diuji | Detail Uji | Jenis Pengujian |
| 1 | Tambah *Certifi* List | User Melakukan tambah sertfikasi milik awak pesawat | *Black Box* |
| 2 | Edit *Certifi* List | User melakukan edit sertifikasi awak pesawat | *Black Box* |
| 3 | *Renewal* *Certifi* | User dengan akses *renewal* ="Y" melakukan pengajuan perpanjangan sertifikasi | *Black Box* |
| 4 | *Verify* 1 | User dengan akses *Verify* 1="Y" melakukan *verify*kasi perpanjangan sertifikasi | *Black Box* |
| 5 | *Verify* 2 | User dengan akses *Verify* 2="Y" melakukan *verify*kasi perpanjangan sertifikasi | *Black Box* |
| 6 | Approval | User dengan akses Approval="Y" melakukan approval perpanjangan sertifikasi | *Black Box* |
| 7 | *Save* Sertifikasi | User dengan akses *save* sertifikasi ="Y" melakukan simpan sertifikasi | *Black Box* |
| 8 | *Save* Bukti Sertifikasi  **Tabel** *4.13 Pengujian Group Information* | User dengan akses *save* bukti sertifikasi ="Y" melakukan simpan bukti sertifikasi | *Black Box* |

### Hasil Pengujian

#### Hasil Pengujian Login/Logout

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Sub Modul | Skenario Uji | Hasil yang diharapkan | Hasil |
| 1 | Login | Login(Jika Benar) | Masuk ke halaman home | Sesuai |
| Login(Jika Salah) | Tampil konfirmasi login gagal | Sesuai |
| 2 | Logout | Logout | Kembali ke menu login | Sesuai |

**Tabel** *4.14 Hasil Pengujian Login/Logout*

#### Hasil Pengujian Group Account

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Sub Modul | Skenario Uji | Hasil yang diharapkan | Hasil |
| 1 | Tambah User | Tambah User(Jika Benar) | Tambah Data User Berhasil | Sesuai |
| Tambah User(Jika Salah) | Tambah Data User Gagal | Sesuai |
| 2 | Edit User | Edit User(Jika Benar) | Edit User Berhasil | Sesuai |
| Edit User(Jika Salah) | Edit User Gagal | Sesuai |
| 3 | Edit Password | Edit Password(Jika Benar) | Edit Password Berhasil | Sesuai |
| Edit Password(Jika Salah) | Password sebelumnya tidak seuai | Sesuai |
| 4 | Edit Hak Akses | Edit Hak Akses(Jika Benar) | Edit Hak Akses Berhasil | Sesuai |
| Edit Hak Akses(Jika Salah) | Edit Hak Akses Gagal | Sesuai |
| 5 | Hapus User | Hapus User(Jika Benar) | Hapus User Berhasil | Sesuai |
| Hapus User(Jika Salah) | Hapus User Gagal | Sesuai |

**Tabel** *4.15**Hasil* *Pengujian* *Group Account*

#### Hasil Pengujian Group Data Support

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Sub Modul | Skenario Uji | Hasil yang diharapkan | Hasil |
| 1 | Tambah Crew *Type* | Tambah Crew *Type*(Jika Benar) | Data Tersimpan di database, data tampil di tabel dan Tampil konfirmasi "Crew *Type* Berhasil ditambahkan" | Sesuai |
| Tambah Crew *Type*(Jika Salah) | Data tidak tersimpan di database dan Tampil Pesan "Code yang sama sudah digunakan" | Sesuai |
| 2 | Tambah Crew *Certifi* | Tambah Crew *Certifi* (Jika Benar) | Data Tersimpan di database, data tampil di tabel dan Tampil Konfirmasi "Tambah Crew *Certifi* berhasil" | Sesuai |
| Tambah Crew *Certifi*(Jika Salah) | Data tidak tersimpan di database dan Tampil Pesan "Code yang sama sudah digunakan" | Sesuai |
| 3 | Tambah *Certifi* *Type* | Tambah *Certifi* Tye(Jika Benar) | Data Tersimpan di database, data tampil di tabel dan Tampil Konfirmasi "Tambah *certifi* *Type* berhasil" | Sesuai |
| Tambah *Certifi* *Type* (Jika Salah) | Data tidak tersimpan di database dan Tampil Pesan "Code yang sama sudah digunakan" | Sesuai |
| 4 | Tambah Crew Profile | Tambah Crew Profile(Jika Benar) | Data Tersimpan di database, data tampil di tabel dan Tampil Konfirmasi "Tambah crew profile berhasil" | Sesuai |
| Tambah Crew Profile(Jika Salah) | Data tidak tersimpan di database dan Tampil Pesan "NIP yang anda masukan sudah ada" | Sesuai |
| 5 | Edit Crew *Type* | Edit Crew *Type*(Jika Benar) | Data di database berhasil di update, data tampil di tabel dan Tampil Konfirmasi "Edit crew *Type* berhasil" | Sesuai |
|  | Edit Crew *Type*(Jika Salah) | Database tidak berhasil di update dan Tampil Pesan "Data sedang digunakan" | Sesuai |
| 6 | Edit Crew *Certifi* | Edit Crew *Certifi*(Jika Benar) | Data di database berhasil di update, data tampil di tabel dan Tampil Konfirmasi "Edit Crew *Certifi* Berhasil" | Sesuai |
|  | Edit Crew *Certifi*(Jika Salah) | Database tidak berhasil di update dan Tampil Pesan "Data sedang digunakan" | Sesuai |
| 7 | Edit *Certifi* *Type* | Edit *Certifi* *Type*(Jika Benar) | Data di database berhasil di update, data tampil di tabel dan Tampil Konfirmasi "Edit *Certifi* *Type* Berhasil" | Sesuai |
|  | Edit *Certifi* *Type*(Jika Salah) | Database tidak berhasil di update dan Tampil Pesan "Data sedang digunakan" | Sesuai |
| 8 | Edit Crew Profile | Edit Crew Profile(Jika Benar) | Data di database berhasil di update, data tampil di tabel dan Tampil Konfirmasi " Edit Crew Profile Berhasil" | Sesuai |
|  | Edit Crew Profile(Jika Salah) | Database tidak berhasil di update dan Tampil Pesan "Data sedang digunakan" | Sesuai |
| 9 | Hapus crew *Type* | Hapus Crew *Type*(Jika Benar) | Data di database terhapus, data di tabel sudah tidak muncul dan Tampil Konfirmasi "Data Berhasil di Hapus" | Sesuai |
|  | Hapus Crew *Type*(Jika Salah) | Data di database tidak terhapus dan Tampil Pesan "Data sedang digunakan" | Sesuai |
| 10 | Hapus Crew *Certifi* | Hapus Crew *Certifi* (Jika Benar) | Data di database terhapus, data di tabel sudah tidak muncul dan Tampil Konfirmasi "Data Berhasil di Hapus" | Sesuai |
|  | Hapus Crew *Certifi*(Jika Salah) | Data di database tidak terhapus dan Tampil Pesan "Data sedang digunakan" | Sesuai |
| 11 | Hapus *Certifi* *Type* | Hapus *Certifi* *Type*(Jika Benar) | Data di database terhapus, data di tabel sudah tidak muncul dan Tampil Konfirmasi "Data Berhasil di Hapus" | Sesuai |
|  | Hapus *Certifi* *Type*(Jika Salah) | Data di database tidak terhapus dan Tampil Pesan "Data sedang digunakan" | Sesuai |

**Tabel** *4.16**Hasil* *Pengujian* *Group Data Support*

#### Hasil Pengujian Group Information

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Sub Modul | Skenario Uji | Hasil yang diharapkan | Hasil |
| 1 | Tambah *Certifi* List | Tambah *Certifi* List(Jika Benar) | Data Tersimpan di database, data tampil di tabel dan Tampil Konfirmasi "Tambah *Certifi* List Berhasil" | Sesuai |
| Tambah *Certifi* List(Jika Salah) | Data tidak tersimpan di database dan Tampil Pesan "*Certifi* yang sama sudah ada" | Sesuai |
| 2 | Edit *Certifi* List | Edit *Certifi* List(Jika Benar) | Data Tersimpan di database, data tampil di tabel dan Tampil Konfirmasi "Edit *Certifi* List Berhasil" | Sesuai |
| Edit *Certifi* List(Jika Salah) | Data tidak tersimpan di database dan Tampil Konfirmasi "Edit *Certifi* List Gagal" | Sesuai |
| 3 | *Renewal* *Certifi* | *Renewal* *Certifi*(Jika Benar) | Data Tersimpan di database, data tampil di tabel dan Muncul Konfirmasi "Data Berhasil diajukan" | Sesuai |
| *Renewal* *Certifi*(Jika Salah) | Data tidak tersimpan di database dan Muncul Pesan "Data sedang proses diajukan" | Sesuai |
| 4 | *Verify* 1 | *Verify*(Jika Benar) | Data Tersimpan di database, data tampil di tabel dan Muncul Konfirmasi "Data Berhasil di *Verify*kasi" | Sesuai |
| *Verify*(Jika Salah) | Data tidak tersimpan di database dan Muncul Pesan "Data Sudah di *Verify*kasi" | Sesuai |
| 5 | *Verify* 2 | *Verify*(Jika Benar) | Data Tersimpan di database, data tampil di tabel dan Muncul Konfirmasi "Data Berhasil di *Verify*kasi" | Sesuai |
| *Verify*(Jika Salah) | Data tidak tersimpan di database dan Muncul Pesan "Data Sudah di *Verify*kasi" | Sesuai |
| 6 | Approval | Approval(Jika Benar) | Data Tersimpan di database, data tampil di tabel dan Muncul Konfirmasi "Data berhasil di *Approve*" | Sesuai |
| Approval(Jika Salah) | Data tidak tersimpan di database dan Muncul Pesan "Data sudah di *Approve*" | Sesuai |
| 7 | *Save* Sertifikasi | *Save* Sertifikasi(Jika Benar) | Data Tersimpan di database, data tampil di tabel dan Muncul Konfirmasi "Data Berhasil di simpan" | Sesuai |
| *Save* Sertifikasi(Jika Salah) | Data tidak tersimpan di database dan "Data gagal disimpan" | Sesuai |
| 8 | *Save* Bukti Sertifikasi | *Save* Bukti Sertifikasi(Jika Benar) | Data Tersimpan di database, data tampil di tabel dan Muncul Konfirmasi "Data Berhasil di simpan" | Sesuai |
| *Save* Bukti Sertifikasi(Jika Salah) | Data tidak tersimpan di database dan "Data gagal disimpan" | Sesuai |

**Tabel** *4.17**Hasil* *Pengujian* *Group Information*

#### Hasil Pengujian Export dan Terima Email

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Sub Modul | Skenario Uji | Hasil yang diharapkan | Hasil |
| 1 | *Export* | *Export*(Jika Benar) | Data yang ter*Export* sesuai dengan permintaan. | Sesuai |
| *Export*(Jika Salah) | Data Blank | Sesuai |
| 2 | Terima Email  **Tabel** *4.18**Hasil* *Pengujian* *Export dan Terima Email* | Terima Email(Jika di Setujui) | Terima Email Persetujuan | Sesuai |
| Terima Email(Jika di Tolak) | Terima Email Penolakan | Sesuai |

## Implementasi (Implementation/Operation)

Implementasi merupakan penerapan hasil akhir pada user setelah melalui tingkatan-tingkatan yang sudah diselesaikan.

### Implementasi Perangkat Lunak

Perangkat lunak yang perlu dipersiapkan dalam implementasi sistem informasi controlling sertifikasi awak pesawat antara lain.

1. Wamp Server

Wamp server digunakan dalam implementasi sistem informasi controlling sertifikasi awak pesawat dikarenakan aplikasi ini sangat *compatible* dengan sistem informasi windows yang akan digunakan sebagai pc server. Selain itu karena penulisan menggunakan bahasa pemrograman PHP yang memanfaatkan framework CodeIgniter serta diintegrasikan dengan database MySQL sehingga Wamp server adalah aplikasi yang sudah mencukupi untuk membackup kebutuhan.

1. Mozilla Firefox

Mozilla Firefox adalah web browser yang digunakan sebagai media akses antarmuka dari sistem informasi controlling sertifikasi awak pesawat ini.

Adapun poin 1*(satu)* yang membutuhkan cukup komputer yang digunakan sebagai server, untuk komputer client cukup mempersiapkan point 2*(dua)* saja ataupun aplikasi sejenisnya yang memiliki fungsi sama.

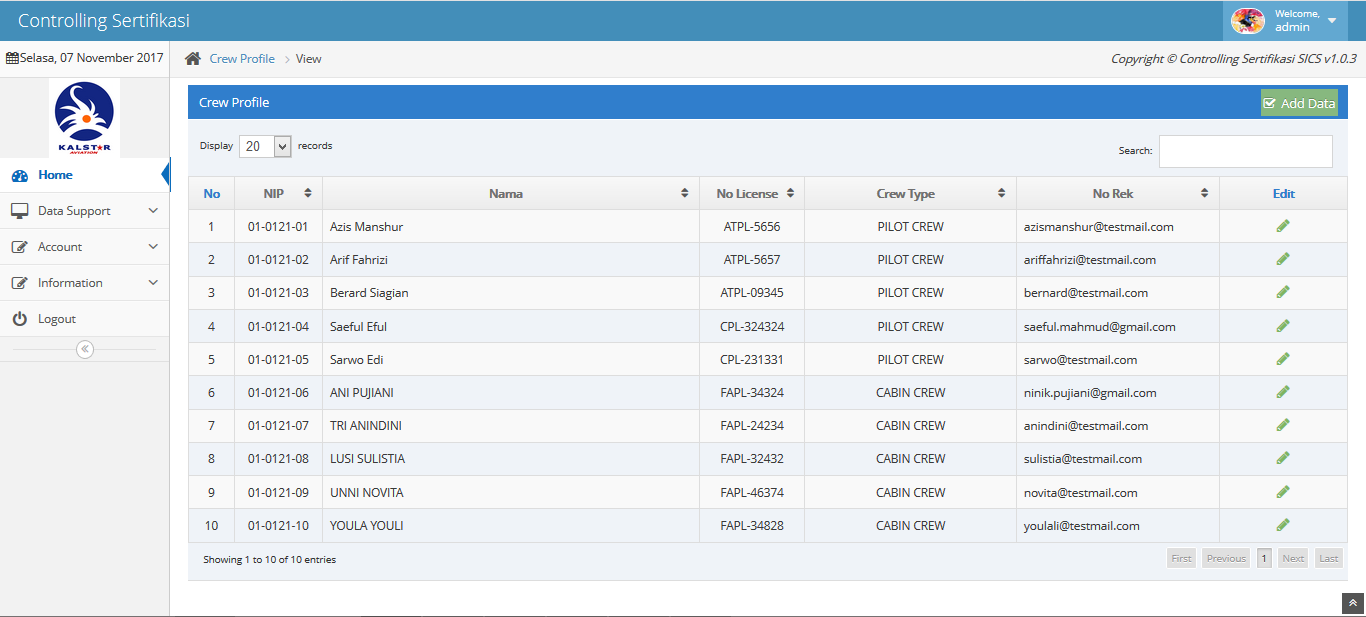
### Implementasi Perangkat Keras

Spesifikasi minimal perangkat keras yang dibutuhkan oleh user dalam penggunaan sistem informasi controlling *certifi*kasi awak pesawat ini sebagai berikut.

1. PC/Laptop Processor Pentium IV
2. Hardisk 80GB
3. RAM 1GB
4. Monitor Color 14”.
5. Mouse dan Keyboard
6. Wifi atau Lan Card

### Hasil Implementasi Antar Muka

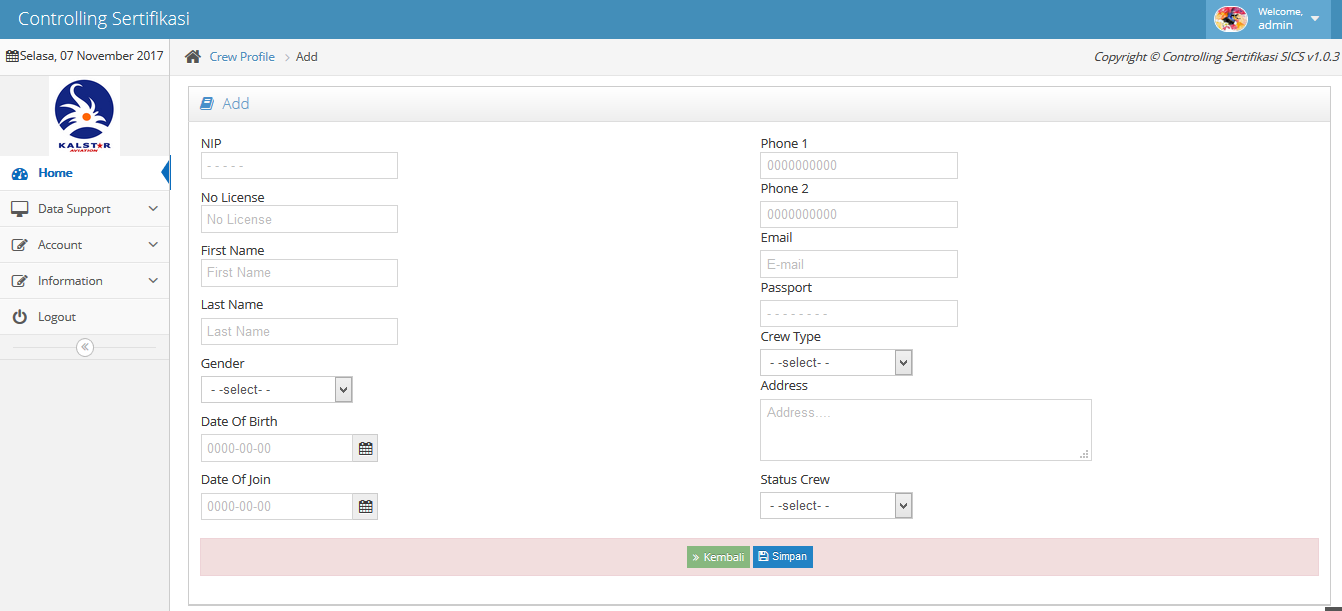
##### Halaman view crew profile



**Gambar** *4.14 Halaman view crew profile*

Keterangan: Menampilkan data crew profile yang dimiliki oleh perusahaan.

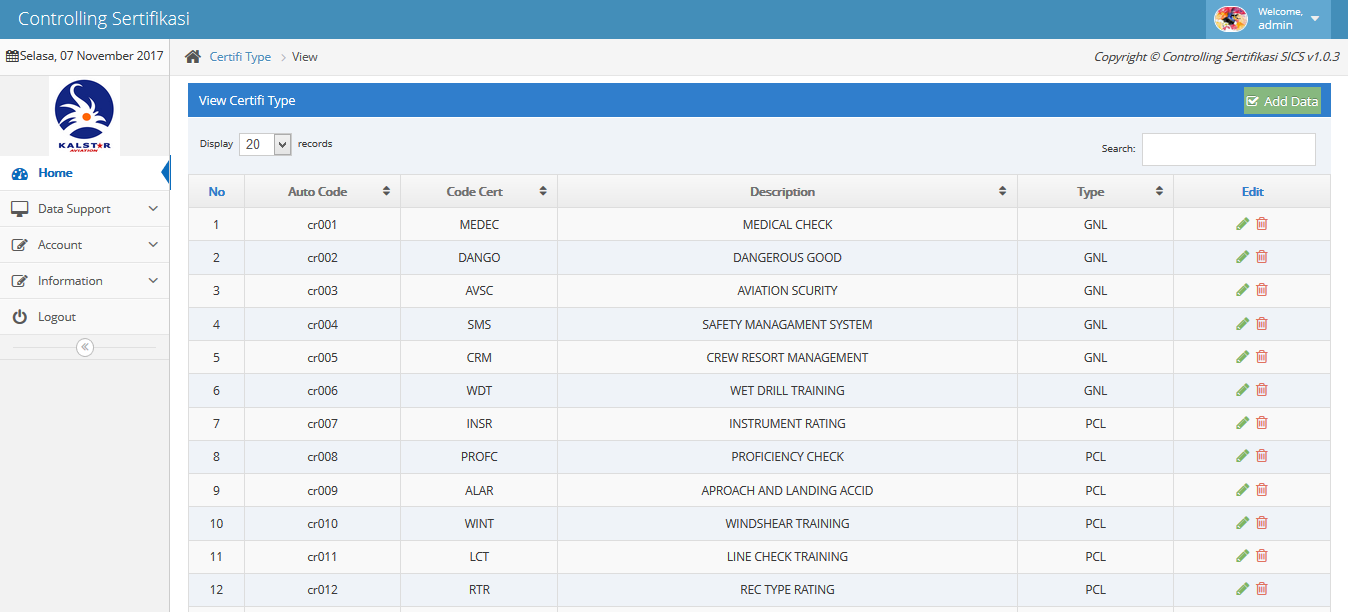
##### Halaman tambah crew profile



***Gambar*** *4.15**Halaman tambah crew profile*

Keterangan: Tambah crew profile digunakan untuk menambah data crew profile yang baru saja bergabung.

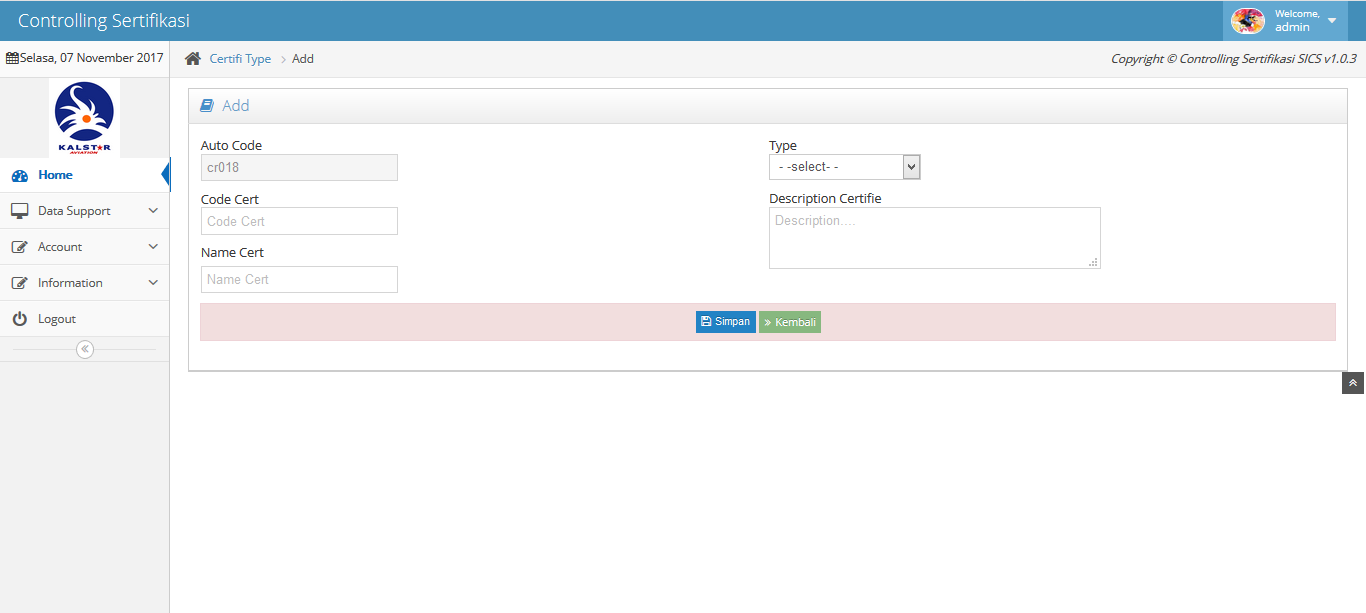
##### Halaman view *certifi* *Type*



**Gambar** *4.16**Halaman view certifi type*

Keterangan: Berfungsi untuk menampilkan data *certifi* awak pesawat.

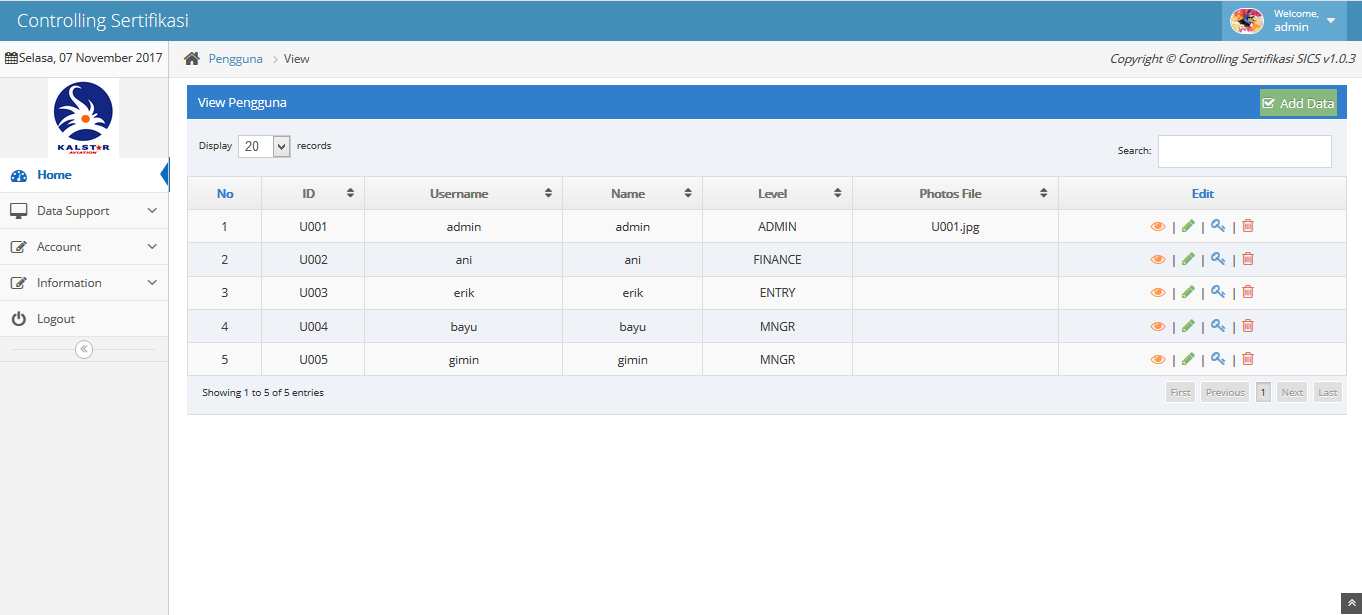
##### Halaman Tambah *certifi* *Type*



**Gambar** *4.17**Halaman tambah certifi type*

Keterangan: Untuk menambahkan data sertifikasi awak pesawat.

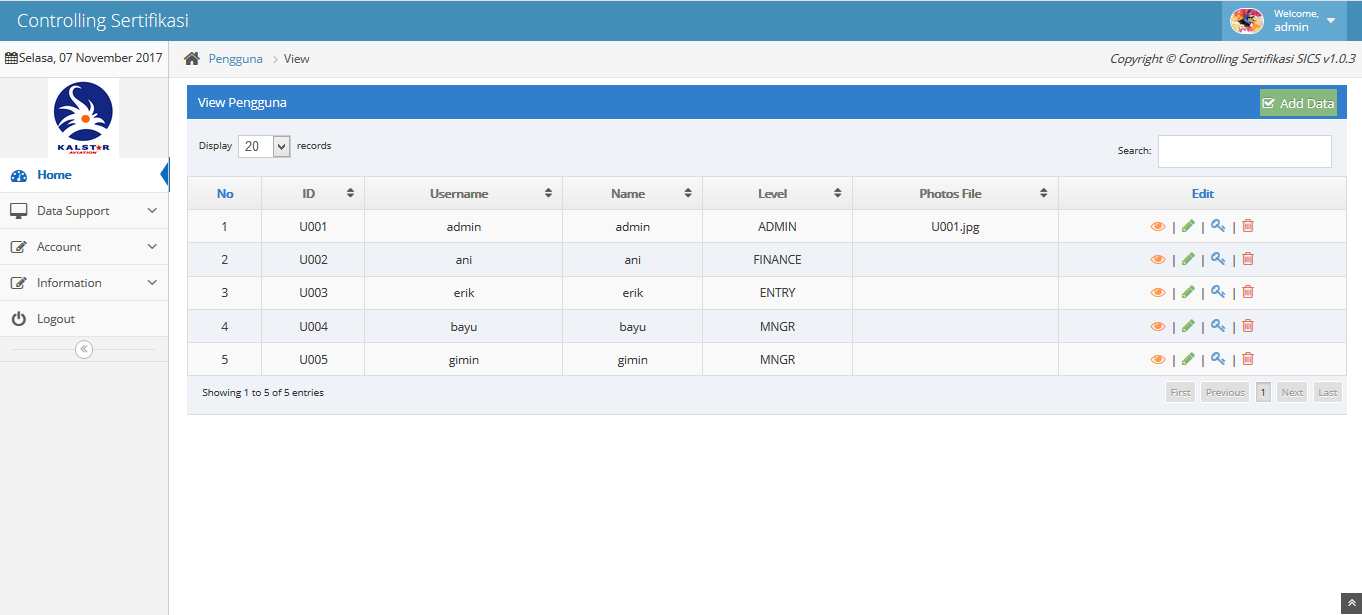
##### Halaman view user account



***Gambar*** *4.18 Halaman view user account*

Keterangan: Untuk menampilkan user yang dapat mengakses aplikasi.

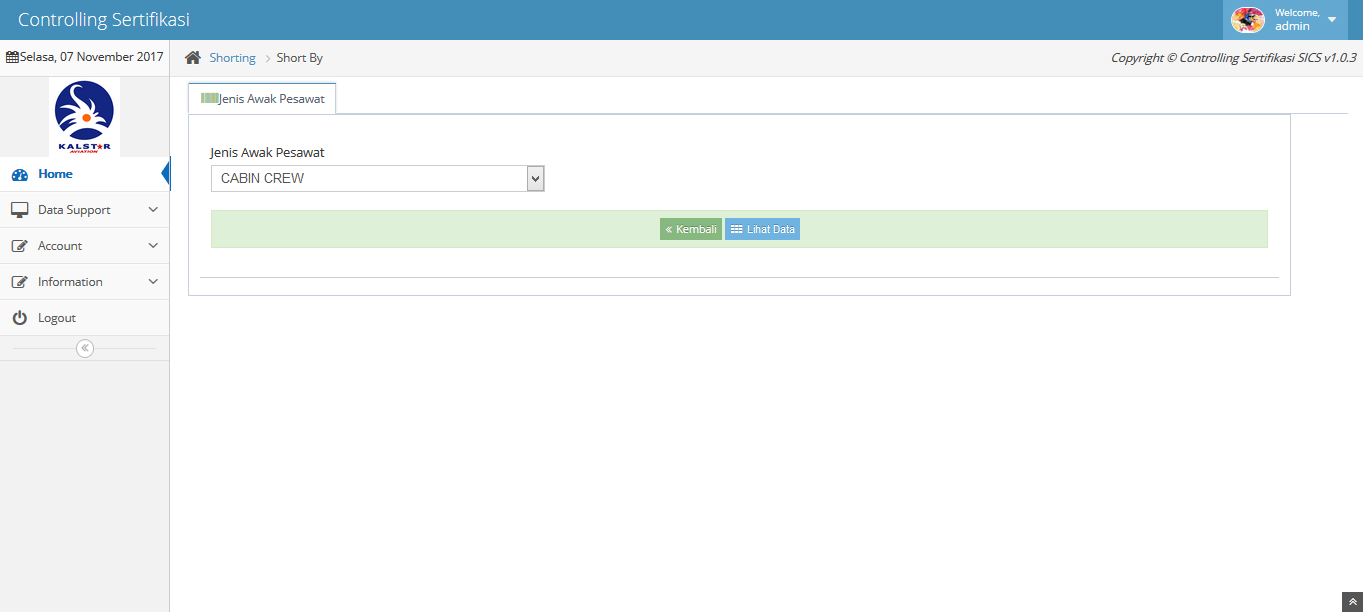
##### Halaman view akses user account



***Gambar*** *4.19**Halaman view akses user account*

Keterangan: menampilkan hak akses yang dimiliki oleh masing-masing user.

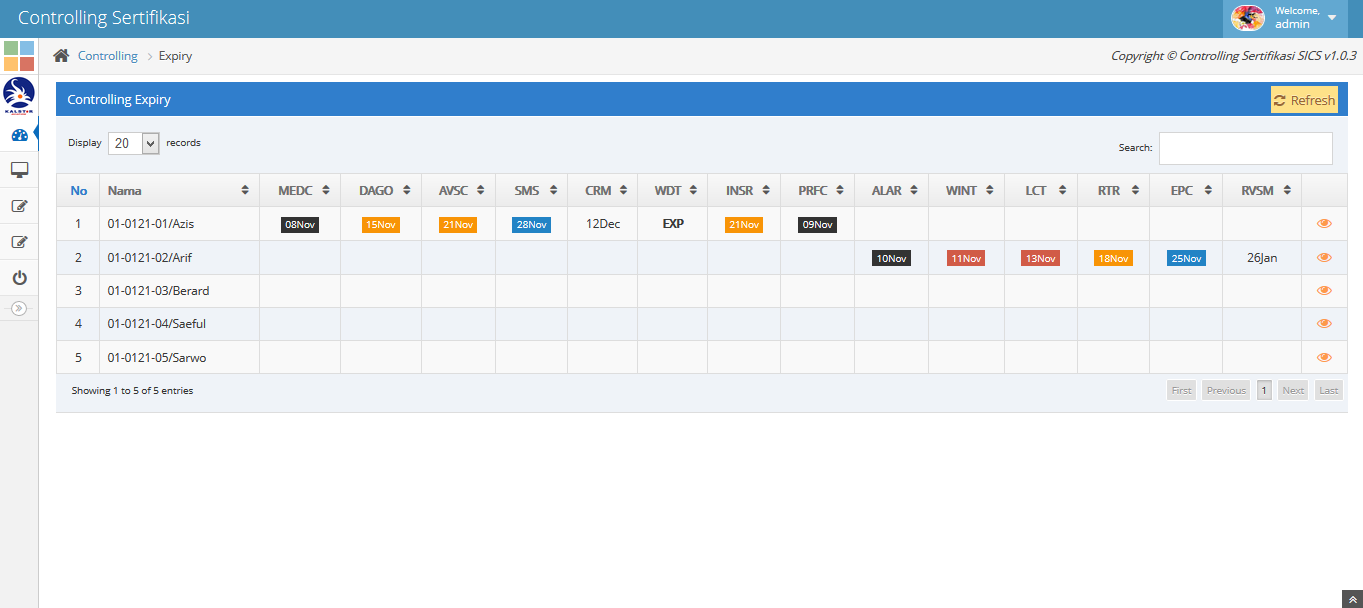
##### Halaman short view *certifi* *Expiry*



***Gambar*** *4.20**Halaman short view controlling expiry*

Keterangan: Untuk memilih jenis awak pesawat yang akan ditampilkan pada tabel sertifikasi *Expiry*.

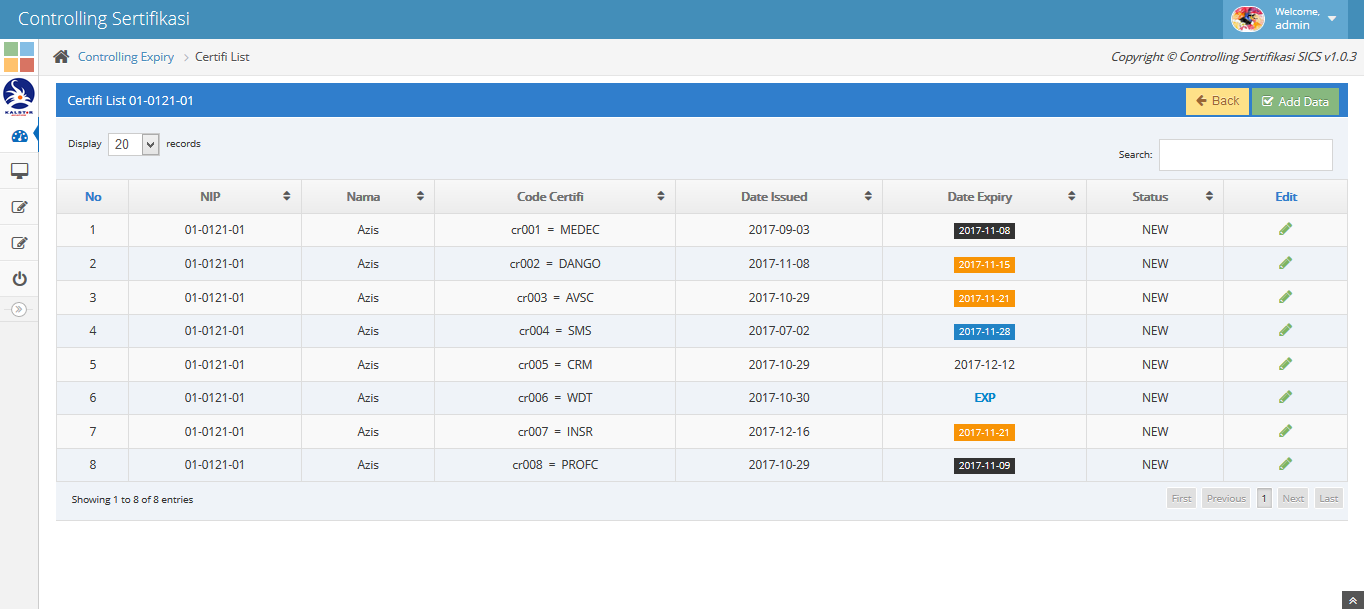
##### Halaman view *Expiry*



***Gambar*** *4.21 Halaman view controlling expiry*

Keterangan: Untuk menampilkan data *Expiry* serta untuk mengontrol data *Expiry* yang dimiliki semua awak pesawat.

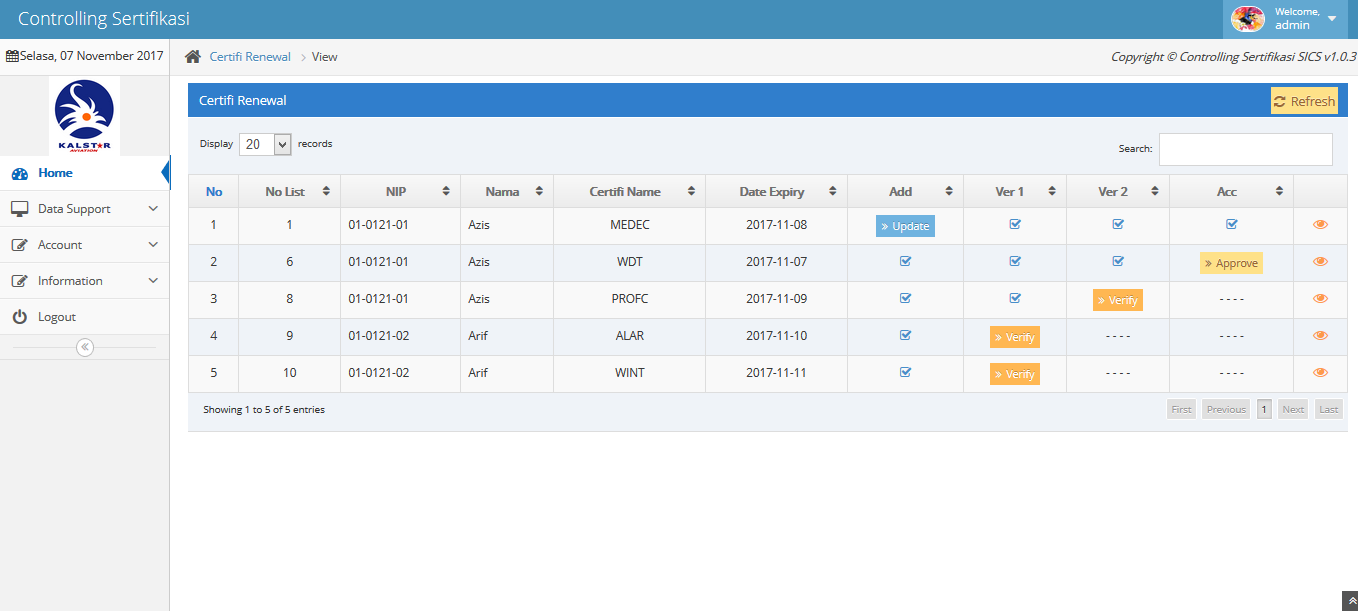
##### Halaman view *certifi* list



***Gambar*** *4.22 Halaman view certifi list*

Keterangan: untuk menampilkan sertifikasi yang dimiliki oleh setiap awak pesawat serta digunakan untuk mengontrol secara detail.

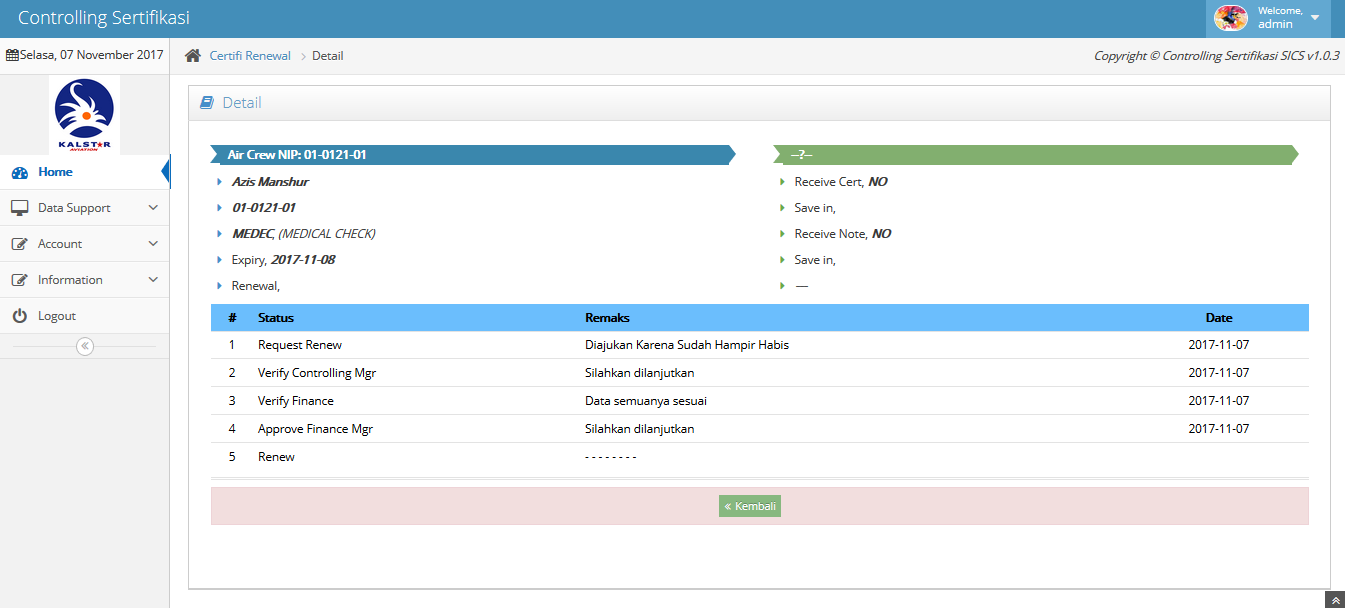
##### Halaman view *certifi* *renewal*



***Gambar*** *4.23 Halaman view certifi renewal*

Keterangan: Berfungsi untuk memonitor sejauh mana proses sertifikasi yang diajukan.

##### Halaman detail *certifi* *renewal*



***Gambar*** *4.24**Halaman view detail certifi renewal*

Keterangan: untuk melihat detail informasi sertifikasi yang diajukan baik itu masukan dari manager, finance ataupun manager finance.

##### Email Pemberitahuan otomatis*(email gateway)*



***Gambar*** *4.25 view email otomatis*

Keterangan: Email pemberitahuan akan otomatis terkirim ke email milik awak pesawat yang saat itu sedang diajukan pembaruan sertifikasinya. Email terkirim saat verifikasi dari manager controlling dan approval dari manager finance.

### Pemeliharaan dan Pengembangan*(Maintenance and Evolution)*

Pemeliharaan dilakukan baik dari sisi sistem ataupun dari hadware komputer. Hal ini dilakukan agar sistem tetap berjalan dengan baik serta agar masalah-masalah yang terjadi pada sistem bisa terdeteksi sehingga tidak menimbulkan problem yang serius.

#### Pemeliharaan Perangkat Lunak

1. Backup data

Langkah backup data sangat penting dilakukan untuk mengantisipasi apabila sewaktu-waktu terjadi masalah pada hadware komputer yang menyimpan data sehingga mengakibatkan data hilang seketika. Lain halnya jika teratur membackup data sehingga apabila terjadi masalah, masih ada data yang bisa digunakan untuk pemulihan.

1. Manajemen Keamanan sistem

Manajemen keamanan sistem juga tidak boleh ditinggalkan mengingat saat ini sedang marak-maraknya aksi penguncian data bahkan pencurian data. Manajemen sistem bisa dengan selalu memperbarui firewal yang dimiliki oleh sistem komputer ataupun menambahkan antivirus dari pihak ke tiga yang di percayakan.

#### Pemeliharaan Perangkat Keras

1. Tempatkan Server di ruangan yang dilengkapi dengan pendingin udara*(Air Conditioner)* bersuhu rendah, atau kalau tidak ada sebisa mungkin server jangan ditempatkan pada tempat yang lembab dan berdebu.
2. Lakukan pembersihan hadware secara berkala agar dapat diketahui sejauh mana hardware tersebut masih mampu bertahan sehingga sudah dipersiapkan hardware penggantinya.

# BAB V

**PENUTUP**

## Kesimpulan

1. Controlling sertifikasi awak pesawat di PT.Kalstar Aviation akan lebih optimal dengan menggunakan sistem informasi controlling sertifikasi awak pesawat ini.
2. Sistem informasi controlling sertifikasi awak pesawat ini memudahkan user penggunanya terutama untuk mengontrol sertifikasi yang akan segera berakhir, melakukan pengajuan pembaruan sertifikasi, melakukan *verify* serta approval sesuai data yang lebih akurat dan dapat memonitor progres perpanjangan sertifikasi yang sudah diajukan baik dari sisi pengguna aplikasi yang dapat memantau langsung melalui sistem, ataupun dari sisi awak pesawat yang selalu mendapatkan update pemberitahuan via email secara otomatis.
3. Sistem ini juga memudahkan user pengguna saat membutuhkan laporan dalam bentuk print out karena data yang di *Export* nantinya tidak perlu untuk dilakukan pengolahan lagi.

## Saran

1. Untuk memudahkan pengolahan data yang berhubungan dengan Controlling sertifikasi, diharapkan ada satu user yang benar-benar menguasai sistem ini agar di kemudian hari apabila timbul pertanyaan dari user yang lain dapat mengerti solusi yang harus dilakukan.
2. Sistem informasi controlling sertifikasi awak pesawat ini diharapkan dapat berkembang lagi, sehingga cangkupan aplikasi tidak hanya sebatas mengontrol sertifikasi awak pesawat tapi juga mengontrol aktifitas yang lain.
3. Untuk memudahkan pengembangan sistem diharapkan user aktif dalam penggunaan sistem ini, dikarenakan tanpa adanya interaksi antara aplikasi dengan user, tidak akan ditemukan satu paham yang sama-sama diharapkan.

# DAFTAR PUSTAKA

Bambang Harianto. *Dasar informatika dan Komputer.* Graha Ilmu. Yogyakarta. 2008

Susanto Azhar. *Sistem Informasi Akuntansi.* Linggar Jaya. Bandung. 2013

Sutarman. *Pengantar teknologi informasi*. Bumi Aksara. Jakarta. 2009

Mulyadi. *Sistem Akuntansi.* Salemba Empat. Jakarta. 2010

Edhy Sutanta ST. *Sistem Informasi Manajemen*. Graha Ilmu. Yogyakarta. 2003

Jogianto. *Analisis dan DesainSistem Informasi*. Edisi II. Yogyakarta. 2004

Abdul Kadir. *Belajar Database Menggunakan MySQL*. Andi Offset. Yogyakarta 2008

Betha Sidik. *Framework CodeIgniter*. Informatika. Bandung. 2012

Sholiq. *Pemodelan Sistem Informasi Berorientasi Obyek dengan UML*. Graha Ilmu. Yogyakarta. 2006

Moh Nazir. *Metode Penelitian*. Ghalia Indonesia. Jakarta. 2005

Azhar Susanto*. Sistem informasi akuntansi konsep dan pengembangan berbasis komputer*. Lingga Jaya. Bandung. 2004

Republic Of Indonesia Ministry of Transportation. *Civil Aviation Safety Regulation(CASR)* Part 121. Amdt-12. Kemenhub: PM61. 2017

Republic Of Indonesia Ministry of Transportation. *Civil Aviation Safety Regulation(CASR)* Part 61. Amdt-5. Kemenhub: PM66. 2017

Referensi Dari Internet

Bobsusanto, *Pengertian Pengawasan Menurut Para Ahli*, Diakses: 19 September 2017 dari: <http://www.spengetahuan.com/2016/10/pengertian-pengawasan-menurut-para-ahli-lengkap.html>

Admin, *Inilah daftar istilah dalam dunia penerbangan*, Diakses: 13November 2017, 07:52WIB dari: <https://airport.id/istilah-dunia-penerbangan/>

# LAMPIRAN

Script Configurasi *(config.php)*

<?php **if** ( ! defined('BASEPATH')) **exit**('No direct script access allowed');

$config['base\_url'] = 'http://' . $\_SERVER['HTTP\_HOST']

. **str\_replace**(basename($\_SERVER['SCRIPT\_NAME']),"", $\_SERVER['SCRIPT\_NAME']);// ini base url

$config['index\_page'] = 'index.php';

$config['nama\_aplikasi'] = 'Controlling Sertifikasi';

$config['nama\_versi'] = 'SICS v1.0.3';

$config['nama\_divisi'] = 'Kalstar Aviation';

Script Configurasi (*Database.php)*

<?php **if** ( ! defined('BASEPATH')) **exit**('No direct script access allowed');

lt']['username'] = 'root';

$db['default']['password'] = '';

$db['default']['database'] = 'db\_sertifikasikontrol';

$db['default']['dbdriver'] = 'mysql';

$db['default']['dbprefix'] = '';

$db['default']['pconnect'] = **TRUE**;

$db['default']['db\_debug'] = **TRUE**;

$db['default']['cache\_on'] = **FALSE**;

$db['default']['cachedir'] = '';

$db['default']['char\_set'] = 'utf8';

$db['default']['dbcollat'] = 'utf8\_general\_ci';

$db['default']['swap\_pre'] = '';

$db['default']['autoinit'] = **TRUE**;

$db['default']['stricton'] = **FALSE**;

1. Bambang Harianto, *Dasar informatika dan Komputer*(Yogyakarta: Graha Ilmu, 2008), h27-28 [↑](#footnote-ref-1)
2. Susanto Azhar,*Sistem Informasi Akuntansi,* (Bandung :Linggar Jaya, 2013), h.22 [↑](#footnote-ref-2)
3. Sutarman, *Pengantar teknologi informasi*, (Jakarta: Bumi Aksara, 2009), h.5 [↑](#footnote-ref-3)
4. Mulyadi,*Sistem Akuntansi,* (Jakarta :Salemba Empat, 2010), h.5 [↑](#footnote-ref-4)
5. Edhy Sutanta ST, *Sistem Informasi Manajemen*,(Yogyakarta: Graha Ilmu, 2003)h.4-6 [↑](#footnote-ref-5)
6. Edhy Sutanta ST, *Sistem Informasi Manajemen*,(Yogyakarta: Graha Ilmu, 2003)h.10 [↑](#footnote-ref-6)
7. Jogianto, *Analisis dan DesainSistem Informasi*, Edisi II,(Yogyakarta, 2004),h.8 [↑](#footnote-ref-7)
8. Susanto Azhar, Sistem Informasi Akuntansi,(Bandung: Linggar Jaya, 2013),h.46 [↑](#footnote-ref-8)
9. Edhy Sutanta ST, *Sistem Informasi Manajemen*,(Yogyakarta: Graha Ilmu, 2003)h.13 [↑](#footnote-ref-9)
10. Bobsusanto, *Pengertian Pengawasan Menurut Para Ahli*, Diakses: 19 September 2017 dari: <http://www.spengetahuan.com/2016/10/pengertian-pengawasan-menurut-para-ahli-lengkap.html> [↑](#footnote-ref-10)
11. Bobsusanto, *Pengertian Pengawasan Menurut Para Ahli*, Diakses: 19 September 2017 dari: <http://www.spengetahuan.com/2016/10/pengertian-pengawasan-menurut-para-ahli-lengkap.html> [↑](#footnote-ref-11)
12. Bambang Harianto, *Dasar informatika dan Komputer*(Yogyakarta: Graha Ilmu, 2008), h232 & 240 [↑](#footnote-ref-12)
13. Bambang Harianto, *Dasar informatika dan Komputer*(Yogyakarta: Graha Ilmu, 2008), h233 & 241 [↑](#footnote-ref-13)
14. Kadir Abdul, *Belajar Database Menggunakan MySQL*,(Yogyakarta: ANDI OFFSET, 2008),h.3 [↑](#footnote-ref-14)
15. Bambang Harianto, *Dasar informatika dan Komputer*(Yogyakarta: Graha Ilmu, 2008), h196-197 [↑](#footnote-ref-15)
16. Kadir Abdul, *Belajar Database Menggunakan MySQL*,(Yogyakarta: ANDI OFFSET, 2008),h.3 [↑](#footnote-ref-16)
17. Admin, *Inilah daftar istilah dalam dunia penerbangan*, Diakses: 13November 2017, 07:52WIB dari: https://airport.id/istilah-dunia-penerbangan/ [↑](#footnote-ref-17)
18. Republic Of Indonesia Ministry of Transportation, *Civil Aviation Safety Regulation(CASR)* Part 121,(Kemenhub: PM61, 2017),Amdt.12 [↑](#footnote-ref-18)
19. Republic Of Indonesia Ministry of Transportation, *Civil Aviation Safety Regulation(CASR)* Part 61,(Kemenhub: PM66, 2017),Amdt.5 [↑](#footnote-ref-19)
20. Bambang Harianto, *Dasar informatika dan Komputer*(Yogyakarta: Graha Ilmu, 2008), h234 [↑](#footnote-ref-20)
21. Bambang Harianto, *Dasar informatika dan Komputer*(Yogyakarta: Graha Ilmu, 2008), h237 [↑](#footnote-ref-21)
22. Kadir Abdul, Belajar Database Menggunakan MySQL,(Yogyakarta: ANDI OFFSET, 2008),h.2 [↑](#footnote-ref-22)
23. Sidik Betha, Framework CodeIgniter,(Bandung: INFORMATIKA, 2012),h.1-2 [↑](#footnote-ref-23)
24. Sholiq, *Pemodelan Sistem Informasi Berorientasi Obyek dengan UML*(Yogyakarta: Graha Ilmu,2006)h.6-7 [↑](#footnote-ref-24)
25. Sholiq, *Pemodelan Sistem Informasi Berorientasi Obyek dengan UML*(Yogyakarta: Graha Ilmu,2006)h.13 [↑](#footnote-ref-25)
26. Sholiq, *Pemodelan Sistem Informasi Berorientasi Obyek dengan UML*(Yogyakarta: Graha Ilmu,2006)h.15-17 [↑](#footnote-ref-26)
27. Sholiq, *Pemodelan Sistem Informasi Berorientasi Obyek dengan UML*(Yogyakarta: Graha Ilmu,2006)h.17-19 [↑](#footnote-ref-27)
28. Sholiq, *Pemodelan Sistem Informasi Berorientasi Obyek dengan UML*(Yogyakarta: Graha Ilmu,2006)h.7 [↑](#footnote-ref-28)
29. Sholiq, *Pemodelan Sistem Informasi Berorientasi Obyek dengan UML*(Yogyakarta: Graha Ilmu,2006)h.8-9 [↑](#footnote-ref-29)
30. Sholiq, *Pemodelan Sistem Informasi Berorientasi Obyek dengan UML*(Yogyakarta: Graha Ilmu,2006)h.14-15 [↑](#footnote-ref-30)
31. Sholiq, *Pemodelan Sistem Informasi Berorientasi Obyek dengan UML*(Yogyakarta: Graha Ilmu,2006)h.9-11 [↑](#footnote-ref-31)
32. Sholiq, *Pemodelan Sistem Informasi Berorientasi Obyek dengan UML*(Yogyakarta: Graha Ilmu,2006)h.11-13 [↑](#footnote-ref-32)
33. Moh Nazir, *Metode Penelitian*(Jakarta:Ghalia Indonesia, 2005), h.175 [↑](#footnote-ref-33)
34. Moh Nazir, *Metode Penelitian*(Jakarta:Ghalia Indonesia, 2005), h.180 [↑](#footnote-ref-34)
35. Azhar Susanto*, Sistem informasi akuntansi konsep dan pengembangan berbasis komputer*,(Bandung: Lingga Jaya, 2004)h.61 [↑](#footnote-ref-35)
36. Edhy Sutanta ST, *Sistem Informasi Manajemen*,(Yogyakarta: Graha Ilmu, 2003)h.122 [↑](#footnote-ref-36)
37. Edhy Sutanta ST, *Sistem Informasi Manajemen*,(Yogyakarta: Graha Ilmu, 2003)h.122 [↑](#footnote-ref-37)
38. Edhy Sutanta ST, *Sistem Informasi Manajemen*,(Yogyakarta: Graha Ilmu, 2003)h.156 [↑](#footnote-ref-38)