

PERANGKAT LUNAK PENGELOLAAN PENDAFTARAN PELATIHAN DI CRMS INDONESIA

Rikky Wisnu Nugraha¹, Martiyanti Widiastuti²

^{1,2}Program Studi Sistem Informasi, Program Studi Sistem Informasi STMIK LPKIA
Jln. Soekarno Hatta No. 456 Bandung 40266, Telp. +62 22 75642823, Fax. +62 22 7564282
Email : ¹r.wisnunugraha@gmail.com, ²martiyanti.widiastuti@fellow.lpkia.ac.id

Abstrak

PT Cipta Raya Mekar Sahitya (Center for Risk Management Studies) disingkat dengan CRMS Indonesia adalah perusahaan yang bergerak dalam bidang pendidikan dan pengembangan dalam bidang Governance, Risk, Compliance dan Sustainability (GRCS). Tahap awal dari proses pelatihan yang diselenggarakan CRMS Indonesia ini adalah melakukan pendaftaran terlebih dahulu, penerbitan surat invoice pada akhirnya terkirimnya sertifikat ke tangan peserta. Namun, dalam kenyataannya bagian sekretariat melakukan hal pengulangan penulisan data pendaftaran yang akan dikirimkan pada email pendaftar, masih belum terpusatnya data, terlalu lamanya dalam pembuatan dokumen invoice pada bagian accounting dan tidak adanya sistem monitoring sertifikat untuk bagian event. Untuk mengatasi masalah tersebut, diperlukan sistem baru yang dapat diakses secara online, penggunaan teknologi informasi dapat memberikan kemudahan akan kebutuhan informasi yang cepat, tepat dan akurat. Metode pengembangan sistem digunakan adalah model *Waterfall* dan penggunaan *Codeigniter Framework* sebagai alat untuk mempermudah dalam pembangunan perangkat lunak. Dari hasil analisa, perancangan dan implementasi serta pengujian perangkat lunak ini dapat diambil kesimpulan bahwa pengelolaan pendaftaran pelatihan dapat mengurangi waktu menjadi lebih efisien seperti pembuatan dokumen invoice maupun pencarian sertifikat peserta. Database yang terpusat membuat tidak adanya perbedaan data antar pegawai dan sistem autorespon pada bagian pendaftaran menjadikan tugas sekretariat menjadi berkurang.

Kata kunci : *Waterfall, Pengelolaan Pendaftaran, CRMS Indonesia, Codeigniter Framework*

1. Pendahuluan

Sumberdaya manusia mempunyai peran besar dan paling penting bagi suatu organisasi, karena unsur manusia dalam suatu organisasi dapat memberikan keunggulan bersaing dengan membuat sasaran, strategi, inovasi dan mencapai tujuan organisasi. Maka salah satu persiapan menghadapi perubahan-perubahan dimasa yang akan datang yaitu dengan mempersiapkan sumberdaya manusia yang berkualitas berupa pelatihan bagi karyawan.

PT Cipta Raya Mekar Sahitya (*Center for Risk Management Studies*) disingkat dengan CRMS Indonesia adalah perusahaan yang bergerak dalam bidang pendidikan dan pengembangan dalam bidang *Governance, Risk, Compliance* dan *Sustainability* (GRCS). Tahap awal dari proses pelatihan yang diselenggarakan CRMS Indonesia ini adalah melakukan pendaftaran terlebih dahulu, penerbitan surat *invoice* pada akhirnya terkirimnya sertifikat ke tangan peserta. Akan tetapi dalam proses bisnis yang berjalan ditemukannya dua kali pengerjaan didalam proses pendaftaran yang mengharuskan bagian sekretariat menulis ulang data pendaftaran dan mengirimkannya kepada *email* pendaftar, lalu ditemukannya juga data-data yang tidak konsisten antar pegawai sehingga menimbulkan kerancuan data, pada bagian pemrosesan pembuatan *invoice* yang dinilai terlalu lama dan juga proses pengarsipan dokumen sertifikat serta pengecekan sertikat yang masih dimasukkan kedalam sebuah kabinet, apabila dokumen tersebut diperlukan akan memerlukan waktu yang lama dikarenakan jumlah dokumen yang

disimpan tidaklah sedikit sehingga dapat menurunkan efisiensi dalam mengerjakan sebuah pekerjaan.

Oleh karena itu, untuk mendukung sistem yang lebih efektif perlu dikembangkan suatu aplikasi pengelolaan pendaftaran peserta pelatihan dengan berbasis web yang lebih mudah digunakan, yang diharapkan dapat menangani masalah-masalah yang timbul didalam proses pengolahan data yang ada di PT Cipta Raya Mekar Sahitya.

Berdasarkan latar belakang masalah di atas, maka dapat diidentifikasi permasalahan sebagai berikut:

1. Masih belum adanya fungsi *auto* konfirmasi pada bagian pendaftaran peserta pelatihan melalui *email* pendaftar. Pencatatan masih dilakukan secara tidak terpusat, sehingga terjadinya ketidak samaan data antar pegawai.
2. Sistem yang ada tidak dapat menentukan no invoice secara otomatis dan harga pelatihan sesuai ketentuan periode pembayaran.
3. Pengecekan sertifikat yang belum dikirim masih dilakukan di sebuah buku sehingga membutuhkan waktu yang lama untuk proses pencariannya.

Agar pembahasan persoalan tidak keluar dari topik yang akan dibahas, adapun ruang lingkup permasalahan sebagai berikut:

1. Sistem ini hanya menangani hal-hal tentang pengelolaan pendaftaran dimulai dari pendaftaran awal, pembayaran dan pembuatan sertifikat peserta.

2. Surat kelengkapan dokumen pembayaran hanya dokumen *invoice*.
3. Pemberitahuan *autorespon* dari pendaftaran dilakukan via *email*

Adapun tujuan pembuatan perangkat lunak pengelolaan pendaftaran pelatihan di CRMS Indonesia ini, sebagai berikut:

1. Mempermudah pekerjaan bagian sekretariat sehingga tidak 2 kali kerja dalam mengirimkan konfirmasi pendaftaran
2. Menciptakan *database* secara terpusat untuk mempermudah para karyawan bagian manajemen dalam pengolahan data, agar seragam satu dengan yang lainnya.
3. Mempercepat bagian *accounting* dalam membuat dokumen *invoice*
4. Mempercepat proses pembuatan ataupun *monitoring* sertifikat yang telah dikeluarkan.

1.1. Landasan Teori

1.1.1. Pengertian Pendaftaran

Menurut Kamus Besar Bahasa Indonesia kata pendaftaran berasal dari kata dasar daftar yang berarti catatan sejumlah nama atau hal yang disusun berderet dari atas ke bawah. Sedangkan arti kata dari pendaftaran adalah merupakan proses atau cara dari perbuatan mendaftar misalnya pencatatan nama, alamat dan sebagainya

1.1.2. Pengertian Pengelolaan

Menurut Harsoyo (2013:1377) dalam jurnal Hendro Pujo yang berjudul Implementasi Pengelolaan dan Penatausahaan keuangan Desa Berdasarkan Perda No 16 Tahun 2007 Tentang Keuangan Desa di Mulawarman Kecamatan Tenggarong Seberang Kabupaten Kutai Kartanegara pengelolaan adalah suatu istilah yang berasal dari kata kelola adalah “serangkaian usaha yang bertujuan untuk menggali dan memanfaatkan segala potensi yang dimiliki secara efektif dan efisien guna mencapai tujuan tertentu yang telah direncanakan sebelumnya”. [2]

1.1.3. Pengertian Pengelolaan Pendaftaran

Menurut kedua pengertian diatas pengertian dari pengelolaan pendaftaran adalah serangkaian tugas ataupun kegiatan yang digunakan untuk mengontrol dan melakukan pengawasan dari proses pencatatan data pribadi peserta hingga tersampainya sertifikat kepada peserta.

1.1.4. Pengertian Pelatihan

Pelatihan (training) adalah

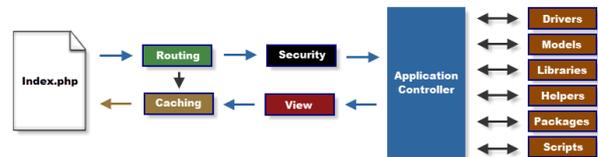
“Suatu proses pendidikan jangka pendek yang mempergunakan prosedur sistematis dan terorganisasi, pegawai non manajerial mempelajari pengetahuan dan

keterampilan teknis dalam tujuan yang terbatas”. (Adrew E.Sikula,2013:117)[4]

1.1.5 Code Igniter Framework

Codeigniter adalah sebuah web application framework yang bersifat open source digunakan untuk membangun aplikasi php dinamis. Tujuan utama pengembangan Codeigniter adalah untuk membantu developer untuk mengerjakan aplikasi lebih cepat daripada menulis semua code dari awal. Codeigniter diperkenalkan kepada publik pada tanggal 28 februari 2006.[3]

CodeIgniter sendiri dibangun menggunakan konsep Model-View-Controller development pattern. Dengan menggunakan pattern MVC ini, struktur kode yang dihasilkan menjadi lebih terstruktur dan memiliki standar yang jelas. Gambar berikut menggambarkan bagaimana data mengalir sepanjang sistem:



Gambar 1. Flow diagram framework CodeIgniter

- 1 Index.php berlaku sebagai controller didepan, yang menginisialisasi sumber dasar yang dibutuhkan untuk menjalankan CodeIgniter
- 2 Router akan memeriksa request HTTP untuk menentukan apa yang harus dilakukan dengan request tersebut.
- 3 Jika ada file cache, maka file cache langsung dikirimkan kepada browser, dengan memotong eksekusi sistem normal
- 4 Keamanan sebelum suatu aplikasi controller dipanggil, request HTTP dan data yang dikirimkan oleh pengguna disaring (filter) terlebih dahulu untuk keamanan.
- 5 Contoller akan memuat model, librari inti, helper dan sumberdaya lainnya yang diperlukan untuk memproses request tersebut
- 6 View terakhir dirender kemudian dikirimkan kepada browser untuk dilihat. Jika caching didefinisikan, maka view akan dicache terlebih dahulu, setelah itu baru dikirimkan kepada browser.

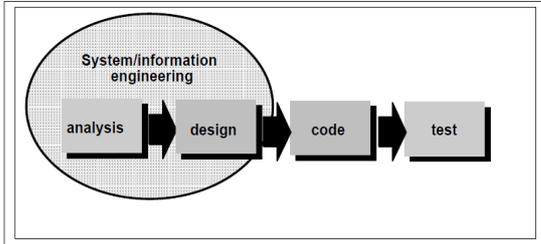
1.1.5 Waterfall

Model *Waterfall* adalah suatu model yang dapat membantu mengatasi kerumitan yang terjadi akibat proyek-proyek pengembangan perangkat lunak. Model *waterfall* memacu tim pengembang untuk merinci apa yang seharusnya perangkat lunak lakukan (mengumpulkan dan menentukan kebutuhan sistem) sebelum sistem tersebut dikembangkan.[1]

Model *Waterfall* adalah model klasik yang bersifat sistematis, berurutan dalam membangun *software* atau perangkat lunak. Berikut ini ada dua gambaran dari

model *waterfall*. Sekalipun keduanya menggunakan nama-nama fase yang berbeda, namun sama dalam intinya.

Fase-fase dalam Model *Waterfall* menurut referensi Pressman:



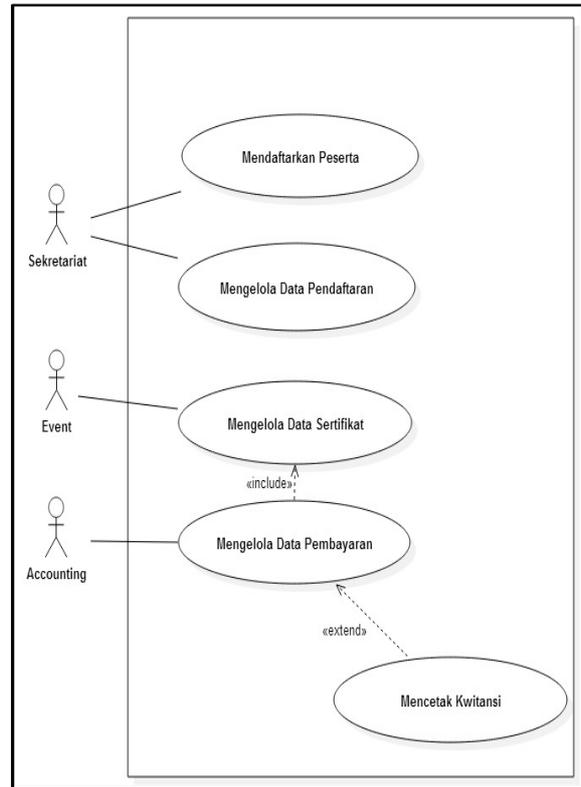
Gambar 2. Model *Waterfall* Pressman

1. Analisis kebutuhan perangkat lunak (*Analysis*) : Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*.
2. Desain (*Design*) : desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean.
3. Pembuatan kode program (*code*) : desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.
4. Pengujian (*Test*) : Pengujian fokus perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji.

2. Gambaran Perangkat Lunak

2.1 Aliran Proses

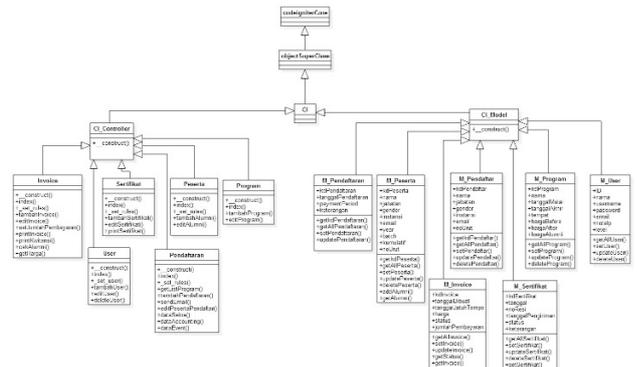
2.1.1 Use Case Diagram



Gambar 3. *Use Case* Diagram

2.2 Class Diagram

Pada *class diagram* ini akan membantu dalam visualisasi struktur class-class dari sistem, *class diagram* memperlihatkan hubungan antar *class* dan penjelasan detail tiap-tiap *class* di dalam model desain dari sistem.

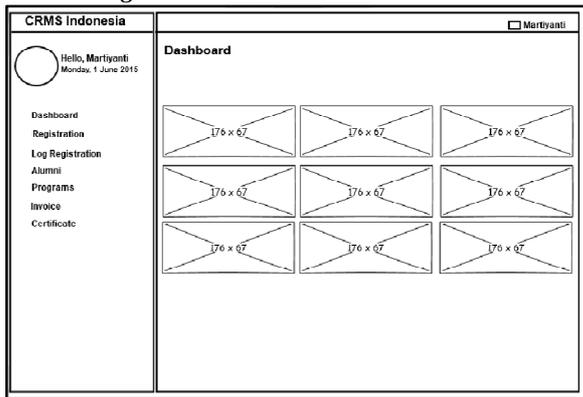


Gambar 4 *Class* Diagram

2.3 Perancangan Antar Muka

Penjabaran komunikasi internal perangkat lunak, antara perangkat lunak dengan sistem luarnya, dan antara perangkat lunak dengan pengguna. Dan ini beberapa perancangan antarmuka yang ada pada perangkat lunak pengelolaan pendaftaran pelatihan.

2.3.1 Dialog ScreenDashboard



Gambar 4 Dialog ScreenDashboard

Uraian penggunaan :

1. Halaman ini adalah halaman utama dari perangkat lunak
2. Halaman ini juga menampilkan jumlah peserta pelatihan dari masing-masing pelatihan yang diselenggarakan.
3. Halaman ini menampilkan menu-menu sesuai fungsionalitasnya
4. Masing-masing menu ditampilkan sesuai level dan jabatan dari *user* tersebut seperti level sekretariat akan memunculkan menu :
 - a. *Registration* digunakan untuk menampilkan halaman pendaftaran peserta pelatihan
 - b. *Log Registration* digunakan untuk menampilkan daftar data pendaftaran
 - c. *Program* digunakan untuk melihat detail dari program yang diselenggarakan
 - d. *Alumni* digunakan untuk melihat data alumni yang telah terdaftar sebelumnya sebagai peserta
 - e. *Certificate* digunakan untuk melihat daftar sertifikat yang telah dikirim ke para peserta
 - f. *Invoice* digunakan untuk melihat data pembayaran *invoice* yang telah dikeluarkan

2.3.2 Dialog ScreenRegistrationForm

Gambar 5 Dialog ScreenRegistrationForm

Uraian penggunaan :

1. Pada halaman ini menampilkan *form* untuk pendaftaran peserta
2. Isi data-data yang ada didalam *form*, lalu tekan tombol *submit* untuk menyimpan data kedalam *database*.

2.3.3 Dialog ScreenCreate Invoice

Gambar 6 Dialog Screen Create Invoice

Uraian Penggunaan :

1. Pada saat *form* ini semua data akan terisi secara otomatis berdasarkan jenis program pelatihan yang diselenggarakan dan jenis pembayaran dari *invoice* tersebut apakah *before event* ataupun *after event*.
2. Lalu klik tombol *submit* untuk menyimpan data *invoice* yang telah dibuat sesuai dengan data yang ada pada *form*.
3. Dan tampilan *invoice* akan ditampilkan.

2.3.4 Dialog Screen Create Certificate

Gambar 7 Dialog Screen Create Certificate

Uraian Penggunaan :

1. Pada saat *form* ini semua data akan terisi secara otomatis berdasarkan nama para peserta dan jenis pelatihan yang diikuti.
2. Lalu klik tombol *submit* untuk menyimpan data sertifikat yang telah dibuat yang telah dibuat.
3. Dan tampilan sertifikat akan ditampilkan.

3.Implementasi

Pada subbab ini menjelaskan tentang langkah-langkah serta rencana jadwal dalam rangka mengimplementasikan perangkat lunak pengelolaan pendaftaran pelatihan yang telah dirancang pada bab sebelumnya.

Daftar Kegiatan

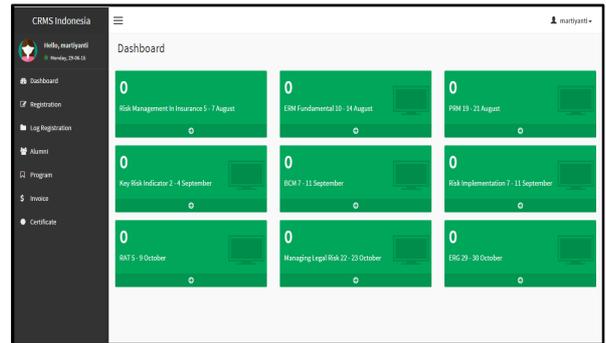
1. Analisis Kebutuhan
Tahap awal menentukan kebutuhan apa saja yang diperlukan untuk membangun perangkat lunak
2. Pengumpulan dan perancangan data
Pengumpulan informasi, pencarian referensi sebagai acua perancangan dan pembangunan perangkat lunak dan mempelajari kebutuhan akan perangkat keras dan perangkat lunak yang akan digunakan untuk pembuatan aplikasi
3. Pembuatan gambar dan desain
Membuat gambar-gambar untuk tampilan anatar muka menu dan menu lainnya yang dipergunakan dalam aplikasi
4. Penulisan kode program
Penulisan kode program berdasarkan module modul yang paling diutamakan
5. Testing dan perbaikan pada aplikasi
Testing terhadap kode program dan perbaikan kesalahan saat program dieksekusi dari kesalahan logika
6. Implementasi penggunaan aplikasi
Penerapan dan pengujian aplikasi apakah ini sesuai harapan atau tidak

3.1 Lingkup dan Batasan Implementasi

Ruang lingkup dan batasan implementasi pengembangan aplikasi ini dilihat dari segi fitur adalah sebagai berikut:

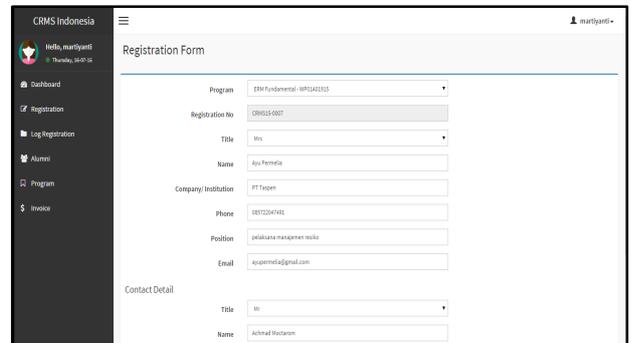
1. Penerapan perangkat lunak pengelolaan pendaftaran pelatihan hanya membahas bagaimana proses pengelolaan pendaftaran peserta, pengelolaan invoice dan tercetaknya sertifikat peserta.
2. Implementasi perangkat lunak pengelolaan pendaftaran hanya membatasi untuk dokumen percetakan hanya sampai tercetaknya sebuah invoice dan sertifikat.

3.2 Implementasi Antarmuka



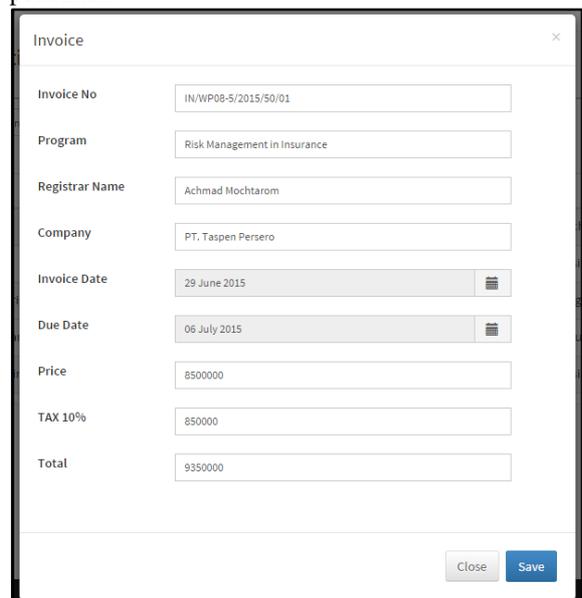
Gambar 8 Dialog Screen Dashboard

Dialog Screen diatas adalah tampilan yang digunakan untuk memberikan informasi jumlah peserta masing-masing program pelatihan



Gambar 9 Dialog Screen Registration Form

Dialog screen diatas adalah form yang digunakan untuk mengisi data diri peserta dan data diri pendaftar pelatihan.



Gambar 10 Dialog Screen Create Invoice

Dialog screen diatas adalah form yang digunakan untuk membuat sebuah invoice pembayaran pelatihan.

Gambar 11 Dialog Screen Create Certificate

Dialog screen diatas adalah form yang digunakan untuk membuat sertifikat peserta pelatihan.

4. Pengujian

Pada *subbab* ini menjelaskan tentang cara, langkah-langkah pengujian aplikasi dan hasil implementasi baik secara modular maupun secara sistem. Pengujian sistem yang akan dilakukan yaitu pengujian *Black Box Alpha* Pengujian tersebut dilakukan untuk mengevaluasi hasil sistem yang dibuat.

Tabel 1 Hasil Pengujian Alpha

No.	Fungsi yang diuji	Cara Pengujian	Hasil pengujian
1.	Login dan hak akses user	Masukkan <i>username</i> dan <i>password</i>	√
2.	Pendaftaran Peserta	Mengisi <i>field</i> yang tersedia di <i>form Registration</i> secara benar lalu tekan tombol <i>submit</i>	√
3.	Validasi form / antarmuka	Menyimpan data masih kosong	√

4.	Melakukan penambahan program pelatihan	Mengisi <i>field</i> yang tersedia di <i>form add program</i> secara benar lalu tekan tombol <i>submit</i>	√
5.	Melakukan pembuatan <i>invoice</i>	Mengisi <i>field</i> yang tersedia di <i>form create invoice</i> secara benar lalu tekan tombol <i>submit</i>	√
6.	Mencetak <i>Invoice</i>	Memilih data <i>invoice</i> yang akan dicetak, klik tombol <i>action</i> pilih <i>printInvoice</i> dan klik tombol <i>print</i>	√
7.	Melakukan pembayaran <i>invoice</i>	Memasukkan jumlah pembayaran <i>invoice</i>	√
8.	Melakukan pembuatan sertifikat	Mengisi <i>field</i> yang tersedia di <i>form create certificate</i> secara benar lalu tekan tombol <i>submit</i>	√
9.	Mencetak sertifikat	Memilih data sertifikat yang akan dicetak, klik tombol <i>action</i> pilih <i>print</i> dan klik tombol <i>print</i>	√

Setelah melalui beberapa tahapan analisa, perancangan dan implementasi, dapat diperoleh kesimpulan sebagai berikut:

1. Dengan adanya fitur *auto* konfirmasi, maka sekretariat tidak perlu menulis ulang data pendaftaran
2. Data menjadi terpusat sehingga jika ada perubahan satu data akan merubah juga ke bagian yang berhubungan dengan data tersebut.

3. Waktu pemrosesan *invoice* dinilai menjadi lebih singkat, efektif dan efisien dikarenakan otomatisnya harga dan nomor *invoice* yang dibuat.
4. Adanya proses pengecekansertifikat sehingga dalam hal pencarian yang telah terkirim dan belum terkirim akan menjadi lebih mudah dan membutuhkan waktu yang tidak lama.

Setelah perancangan aplikasi ini selesai, ada beberapa saran yang akan dikemukakan untuk melengkapi kekurangan dari aplikasi ini, saran-saran tersebut antara lain sebagai berikut:

1. Untuk pemberitahuan informasi pelatihan, sistem yang mengirimkannya secara otomatis via *email*, layaknya *email* marketing yang ditujukan kepada para peserta pelatihan.
2. Ditambahkannya pembuatan *softcopy* dari setiap sertifikat yang telah dicetak.
3. Menambahkannya proses absensi pelatihan untuk peserta pelatihan.

DAFTAR PUSTAKA

- [1]A.S , Rosa dan M.Shalahuddin. 2013 *.Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek* .Bandung :Informatika
- [2]Hendro Pujo Sasongko Adi. 2013 *.Implementasi Pengelolaan dan Penatausahaan keuangan Desa Berdasarkan Perda No 16 Tahun 2007 Tentang Keuangan Desa di Mulawarman Kecamatan Tenggarong Seberang Kabupaten Kutai Kartanegara* .Jakarta: EJournal Ilmu Pemerintahan. Volume 1, Nomor 4, 2013
- [3]Hidayatullah, Priyanto dan Jauhari Khairul Kawistara. 2014 *.Pemrograman Web* .Bandung :Informatika
- [4]Suwanto dan Donni Juni Priansa. 2013. *Manajemen SDM dalam organisasi public dan bisnis*. Bandung :Alfabeta