

Rancang Bangun Aplikasi *Manpower Request* Bagian *Recruitment* pada PT Dynapack Asia

Listiani¹, Friska Natalia²

Program Studi Sistem Informasi, Universitas Multimedia Nusantara, Tangerang Selatan, Indonesia
listiani@student.umn.ac.id
friska.natalia@umn.ac.id

Diterima 24 Januari 2019

Disetujui 24 Juni 2019

Abstract— PT Dynapack Indonesia consists of several divisions, one of them is IT division which has Application Development sub division that is in charge of designing and developing application which is needed for the company. PT Dynapack Indonesia is a multinational company, that it is necessary to maintain the quality of Human Resources, considering the Human Resources (HR) division is a strategic factor for both profit organizations and non-profit organizations [1]. Therefore, the IT division of PT Dynapack Indonesia developed a web-based Manpower Request application system with the purpose of assisting HR division in recruiting process to data archiving. Development of Manpower Request application system uses CodeIgniter framework, jQuery, Metronic, Bootstrap, SAP and MySQL database. Testing, fixing bugs, and updates has been done while the application was being developed. Manpower Request application is ready to use and has been made in accordance with requirements.

Keywords — CodeIgniter, Human Resources, Recruitment, website

I. PENDAHULUAN

Seiring berkembangnya teknologi, pencarian terhadap sesuatu sangat bergantung pada teknologi dan berdasarkan riset sebelumnya ada aplikasi yang digunakan untuk menemukan alergi dalam bentuk website [2] dan untuk menemukan rute terpendek yang dibuat berbasis android [3]. Pada penelitian ini akan dibuat suatu aplikasi berbasis website untuk meningkatkan efisiensi dalam rekrutmen dan pencatatan kebutuhan tenaga kerja.

PT Dynapack Asia merupakan perusahaan *rigid packaging* yang memiliki 10 perusahaan serta 20 pabrik manufaktur di 5 negara (Indonesia, Thailand, Vietnam, Malaysia, dan China). PT Dynapack Indonesia merupakan salah satu bagian dari Dynapack Asia yang bertempat di Indonesia. Perusahaan ini menggunakan SAP untuk melakukan integrasi antar perusahaan dalam Dynapack Asia.

PT Dynapack Indonesia telah menggunakan teknologi informasi untuk membantu proses bisnis yang berjalan, namun masih ada beberapa hal yang berjalan secara manual, seperti: proses rekrutmen calon pekerja dan proses permintaan tambahan pekerja. Pada perusahaan multinasional seperti Dynapack, diperlukan efisiensi kinerja dalam setiap proses bisnis. Agar dapat meningkatkan efisiensi dalam rekrutmen dan pencatatan kebutuhan tenaga kerja, divisi *Human Resources* pada PT Dynapack Indonesia memutuskan untuk melakukan automasi proses rekrutmen dengan membuat sistem aplikasi rekrutmen berbasis web.

II. LANDASAN TEORI

A. Pemrograman Berorientasi Objek

Pemrograman berorientasi objek merupakan praktik menciptakan arsitektur perangkat lunak yang memungkinkan fleksibilitas melalui desain yang bersifat modular.

Pemrograman berorientasi objek bukan merupakan bahasa, melainkan merupakan konsep berpikir dibalik pengembangannya [2].

B. PHP

PHP merupakan bahasa *server-side* yang didesain untuk membangun aplikasi berbasis website. PHP telah digunakan oleh beberapa aplikasi manajemen konten bisnis online, selain itu juga merupakan alat pengembangan situs web dinamis. Kode PHP dapat disisipkan dengan mudah di dalam bahasa HTML. PHP mendukung banyak basis data, seperti: MySQL, Generic ODBC, Solid, dan Oracle [4].

III. METODE PENELITIAN

A. Metode Pengumpulan Data

Data yang dibutuhkan serta alur sistem didapatkan melalui observasi serta wawancara di Dynapack Asia pada tanggal 3 Juli 2017 hingga 5 Juli 2017.

B. Metode Pengembangan Sistem

Implementasi model akan dibangun dengan menggunakan metode *Rapid Application*

Development. Berikut ini merupakan tahapan pengembangan sistem yang digunakan:

1. Analisa dan Perancangan Sistem

Pada tahap ini, dilakukan wawancara dengan divisi *Human Resource* serta observasi langsung untuk mengetahui masalah dan proses bisnis yang terjadi pada proses rekrutmen dan proses permintaan karyawan baru.

Setelah mengetahui proses bisnis yang berjalan serta masalah yang perlu diselesaikan, dilakukan analisa dan perancangan sistem yang dapat menyelesaikan masalah yang ada.

2. Pengembangan Antar Muka

Antar muka aplikasi dikembangkan berdasarkan analisa sistem yang telah dilakukan pada tahap sebelumnya. Antar muka dikembangkan dengan menggunakan Axure RP 7.

3. Implementasi

Basis data serta sistem fungsional dibangun di tahap ini. Sistem fungsional serta basis data diimplementasikan terhadap antar muka yang telah dikembangkan pada tahap sebelumnya. Implementasi sistem fungsional dilakukan menggunakan bahasa pemrograman PHP dan kerangka kerja CodeIgniter

4. Pengujian dan Pemeliharaan

Pengujian dilakukan untuk memastikan bahwa program berjalan sesuai yang diharapkan. Setelah lolos tahap pengujian, sistem aplikasi Manpower Request dapat digunakan oleh seluruh karyawan Dynapack Asia.

Pemeliharaan aplikasi dilakukan agar aplikasi Manpower Request tetap relevan mengikuti perkembangan teknologi dan proses bisnis.

5. Dokumentasi

Dokumentasi merupakan bagian yang penting dalam proses pengembangan sistem aplikasi. Dan yang paling penting, menghilangkan mendokumentasikan proses pengembangan dapat menyebabkan hilangnya data perusahaan [5].

IV. HASIL DAN PEMBAHASAN

A. Analisa dan Perancangan Sistem

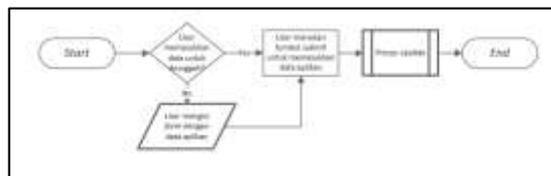
Hasil analisa didapkatans setelah wawancara dengan divisi *Human Resource* serta observasi langsung untuk mengetahui masalah dan proses bisnis yang terjadi pada proses rekrutmen dan proses permintaan karyawan baru. Oleh karena itu, sebelum masuk ke dalam pengembangan sistem, diperlukan alur sistem yang jelas agar pengembangan sistem tidak terhambat. Sistem aplikasi rekrutmen berbasis *web* yang selanjutnya akan disebut *Manpower Request* bagian rekrutmen memiliki beberapa fitur yaitu:

- Fitur *Create Request*
- Fitur *Update Request*
- Fitur *History Request*
- Fitur *Request Next List*
 - Fitur *Pretest*
 - Fitur *HR Interview*

- Fitur *Psychological Test*
- Fitur *User Interview*
- Fitur *Offering*
- Fitur *Medical Checkup*
- Fitur *Finalize*

A.1. Fitur *Create Request*

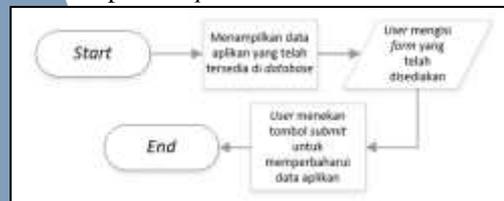
Pengguna dapat membuat permintaan untuk tenaga kerja baru. Setelah divisi *Human Resources* menerima permintaan, divisi *Human Resources* akan memutuskan apakah permintaan ini layak untuk diterima.



Gambar 1. Flowchart Fitur *Create Request*

A.2. Fitur *Update*

Pengguna yang telah membuat permintaan dapat mengubah dokumen permintaan yang sudah diajukan selama divisi *Human Resources* belum melakukan proses *update*.



Gambar 2. Flowchart Fitur *Create Request*

A.3. Fitur *History Request*

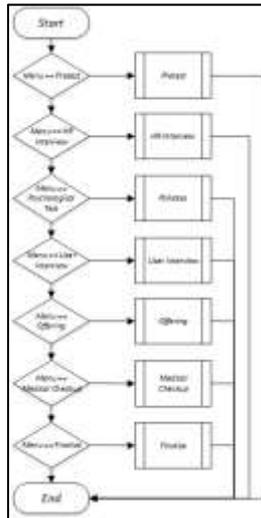
Pengguna yang telah membuat permintaan dapat memantau status permintaan yang telah diajukan.



Gambar 3. Flowchart Fitur *History Request*

A.4. Fitur Request Next List

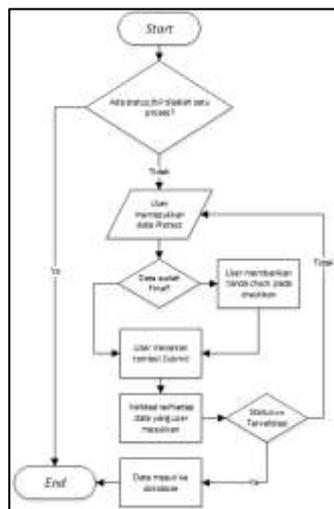
Apabila permintaan disetujui oleh divisi *Human Resources*(HR), kini permintaan akan muncul di halaman *Next List* sehingga HR dapat memulai proses rekrutmen.



Gambar 4. Flowchart Fitur Request Next List

1. Fitur Pretest

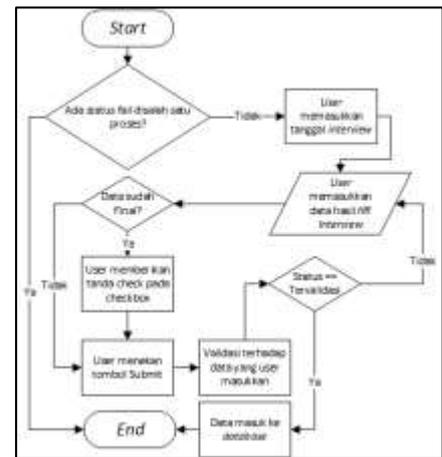
Pada fitur ini, pengguna dapat memasukkan hasil *Pretest* aplikasi. Sistem akan memeriksa apakah ada status gagal disalah satu proses *HR Interview*, *Psikotest*, *User Interview*, *Offering*, *Finalize*. Jika ditemukan kegagalan disalah satu proses, pengguna tidak dapat menambahkan data *Pretest*. Setelah pengguna menambahkan data *Pretest*, pengguna memutuskan apakah data yang dimasukkan sudah bersifat final. Jika dianggap sudah final, maka pengguna dapat memberikan tanda *check* pada kotak *check* yang tersedia. Kemudian, pengguna menekan tombol *Submit* untuk melakukan validasi data kemudian memasukkan data ke dalam *database*.



Gambar 5. Flowchart Fitur Pretest

2. Fitur HR Interview

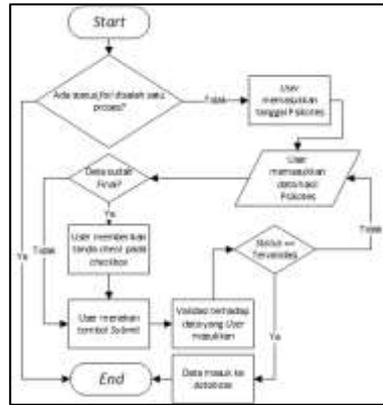
Pada fitur ini, pengguna dapat memasukkan hasil *HR Interview* aplikasi. Sistem akan memeriksa apabila ada status gagal di salah satu proses. Jika tidak ada, pengguna dapat memasukkan tanggal *Interview* oleh HR. Setelah HR melakukan *interview* dengan aplikasi, pengguna akan memasukkan data hasil *interview* HR. Apabila data yang dimasukkan sudah final, pengguna memberikan tanda *check* pada *checkbox* yang tersedia. Setelah mengisi data, pengguna menekan tombol *Submit* dan sistem akan melakukan validasi terhadap data yang dimasukkan oleh pengguna. Apabila data lolos proses validasi, data akan masuk ke *database*.



Gambar 6. Flowchart Fitur HR Interview

3. Fitur Psychological Test

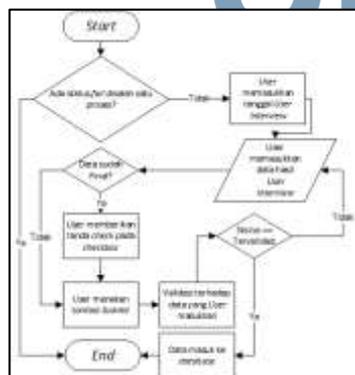
Pada fitur ini, pengguna dapat memasukkan hasil *Psikotes* aplikasi. Pada bagian *Psikotes*, sistem akan memeriksa apabila ada status gagal di salah satu proses. Jika tidak ada, pengguna dapat memasukkan tanggal *Psikotes*. Setelah aplikasi melakukan *psikotes*, data hasil tes perlu dilampirkan ke sistem. Apabila data yang dimasukkan sudah final, pengguna memberikan tanda *check* pada *checkbox* yang tersedia. Tanda *checkbox* berarti data sudah tidak dapat diubah lagi. Setelah mengisi data, pengguna menekan tombol *Submit* dan sistem akan melakukan validasi terhadap data yang dimasukkan oleh pengguna. Apabila data lolos proses validasi, data akan masuk ke *database*.



Gambar 7. Flowchart Pretest

4. Fitur User Interview

Pada fitur ini, pengguna dapat memasukkan hasil *User Interview* aplikasi. Pada bagian *User Interview*, sistem akan memeriksa apabila ada status gagal di salah satu proses. Jika tidak ada, pengguna dapat menjadwalkan tanggal untuk melakukan *User Interview*. Sistem akan mengirimkan *e-mail* ke *user* yang akan melakukan *interview* tiga hari dan satu hari sebelum tanggal *interview* yang dijadwalkan. *User* merupakan calon atasan dari aplikasi tersebut saat ia bekerja. Setelah *User* melakukan *interview* dengan aplikasi, seluruh data hasil *interview* perlu dilampirkan dalam sistem. Apabila data yang dimasukkan sudah final, pengguna memberikan tanda check pada checkbox yang tersedia. Setelah mengisi data, pengguna menekan tombol *Submit* dan sistem akan melakukan validasi terhadap data yang dimasukkan oleh pengguna. Apabila data lolos proses validasi, data akan masuk ke database.

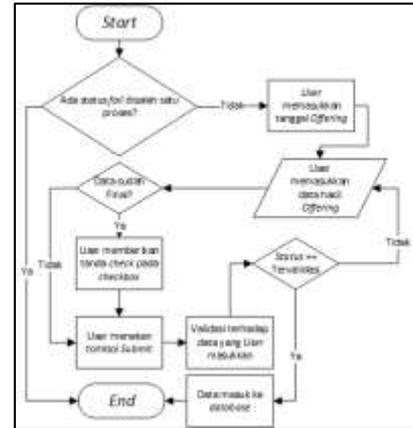


Gambar 8. Flowchart User Interview

5. Fitur Offering

Pada fitur ini, pengguna dapat memasukkan hasil *Offering* aplikasi. Sistem akan memeriksa apabila ada status gagal di salah satu proses. Jika tidak ada, pengguna dapat memasukkan tanggal *Interview* oleh HR. Setelah HR melakukan *interview* dengan aplikasi, pengguna akan memasukkan data hasil *interview* HR. Apabila

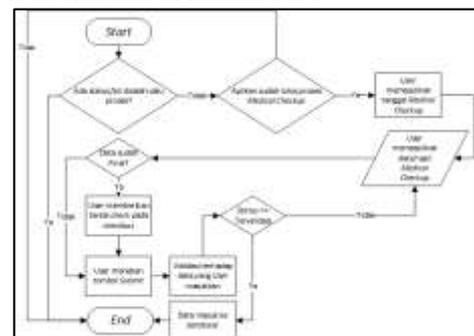
data yang dimasukkan sudah final, pengguna memberikan tanda *check* pada *checkbox* yang tersedia. Setelah mengisi data, pengguna menekan tombol *Submit* dan sistem akan melakukan validasi terhadap data yang dimasukkan oleh pengguna. Apabila data lolos proses validasi, data akan masuk ke *database*.



Gambar 9. Flowchart Offering

6. Fitur Medical Checkup

Pada fitur ini, pengguna dapat memasukkan hasil *Medical Checkup* aplikasi. Sistem memeriksa apabila ada status gagal di salah satu proses. Selain itu, sistem juga memeriksa apakah aplikasi sudah lolos tahap *offering*. Jika sudah lolos tahap *offering* dan belum ada status gagal pada seluruh proses, pengguna dapat memasukkan tanggal *Medical Checkup*. Setelah aplikasi menyelesaikan *medical checkup*, pengguna melampirkan dokumen tes aplikasi. Apabila data yang dimasukkan sudah final, pengguna memberikan tanda check pada checkbox yang tersedia. Setelah mengisi data, pengguna menekan tombol *Submit* dan sistem akan melakukan validasi terhadap data yang dimasukkan oleh pengguna. Apabila data lolos proses validasi, data akan masuk ke database.



Gambar 10. Flowchart Medical Check-up

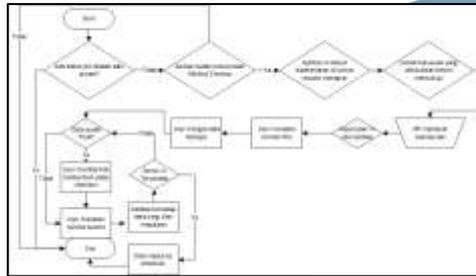
7. Fitur Finalize

Pada fitur ini, pengguna dapat memasukkan hasil *Finalize* aplikasi. Sistem akan

melakukan pemeriksaan, apakah aplikasi yang bersangkutan telah memenuhi ketentuan yang dibutuhkan. Agar seorang aplikasi dapat di-hire, aplikasi perlu memenuhi syarat berikut ini:

- Tidak ada status gagal di salah satu proses
- Aplikasi sudah lolos *Medical Checkup*
- Aplikasi belum dipekerjakan di nomor *request* manapun
- Jumlah karyawan yang dibutuhkan belum mencukupi

Kemudian, divisi *Human Resources* (HR) akan menganalisa dan membuat pilihan untuk menolak aplikasi atau menerima aplikasi. Apabila HR memutuskan untuk menerima aplikasi, pengguna akan mengisi beberapa data kemudian menekan tombol *Hire*, setelah data tervalidasi, data akan masuk ke dalam *database*.



Gambar 11. Flowchart Finalize

B. Pengembangan Antar Muka

Setelah mengetahui secara jelas alur sistem, dilakukan pengembangan antar muka untuk setiap proses.

Gambar 12 menunjukkan antar muka halaman untuk membuat permintaan tenaga kerja baru.



Gambar 12. Antar Muka Halaman *Create Request*

Gambar 13 menunjukkan antar muka halaman untuk menampilkan *history* permintaan tenaga kerja yang telah dibuat.



Gambar 13. Antar Muka Halaman *History Request*

Gambar 14 menunjukkan antar muka halaman *Recruitment – Next List* yang digunakan untuk melihat *Request* yang telah mendapatkan persetujuan dari pihak yang bersangkutan dan tersedia untuk melakukan proses *hiring*.



Gambar 14. Antar Muka Halaman *Next List*

Gambar 15 merupakan antarmuka halaman *Pretest*, *HR Interview*, *Psychological Test*, *User Interview*, *Offering*, serta *Medical Checkup*. Keenam bagian tersebut akan memiliki antarmuka yang sama. Pengguna dapat menjadwalkan serta menyimpan data hasil proses yang bersangkutan.



Gambar 15. Antar Muka Halaman *Pretest*

Gambar 16 merupakan antarmuka halaman *Finalize* yang digunakan untuk menentukan apabila kandidat tersebut lolos dan berhak menjadi karyawan Dynapack Asia.



Gambar 16. Antar Muka Halaman *Finalize*

C. Implementasi

Implementasi dilakukan dengan menggunakan bahasa pemrograman PHP dengan framework CodeIgniter, serta menggunakan MySQL sebagai database.

D. Testing dan Maintenance

Testing dilakukan untuk memastikan bahwa sistem yang dibangun telah bebas dari kesalahan dan sesuai dengan permintaan *client*. Dalam kasus ini, divisi *Human Resources* adalah sebagai user dari testingnya. Proses *maintenance* juga masih berlangsung hingga sekarang. Tahap terakhir adalah melakukan perpindahan dari sistem yang lama ke sistem yang baru. Sebelum adanya *Manpower Request*, seluruh proses bersifat manual atau hanya dilakukan pertukaran dokumen secara fisik. Proses elektronik hanya terjadi pada saat proses hiring. Divisi *Human Resources* akan memasukkan data tenaga kerja baru ke dalam sistem SAP.

Dengan adanya *Manpower Request* pertukaran dokumen secara fisik dapat diganti menjadi pertukaran informasi secara elektronik dan lebih mudah dalam melakukan proses integrasi antar perusahaan manufaktur serta menghindari redundansi data. Redundansi data yang dimaksud adalah jika calon tenaga kerja A pernah melamar di salah satu perusahaan Dynapack, kemudian ditolak, namun suatu saat ketika ia melamar ulang, bagian *Human Resources* tidak perlu memasukkan datanya lagi. Sistem akan mendeteksi serta memberikan notifikasi bahwa orang tersebut sudah ada di dalam *database*.

Konversi sistem terdiri dari empat pendekatan, yaitu: pendekatan langsung, paralel, *phase-in*, serta pilot [5]. Pendekatan yang digunakan pada studi kali ini merupakan pendekatan paralel, yaitu sistem lama tetap digunakan sambil menggunakan sistem baru.

E. Dokumentasi

Pada tahap ini dilakukan penulisan dokumentasi mengenai fungsi, alur proses, serta *user guide* yang digunakan pada saat dilakukan sosialisasi mengenai sistem aplikasi *Manpower Request*.

V. SIMPULAN

A. Kesimpulan

Aplikasi *Manpower Request* berbasis *web* bagian *Recruitment* telah berhasil dirancang dan dibangun pada PT Dynapack Indonesia. Aplikasi ini dibangun dengan *framework* CodeIgniter dan memiliki beberapa fitur, yaitu fitur penambahan data aplikasi menggunakan form *excel*, fitur *reporting*, fitur *Next Process* untuk mengakomodir proses tes yang akan dilalui aplikasi hingga akhirnya divisi *Human Resources* dapat melakukan proses *hiring* pada aplikasi tersebut. Aplikasi ini dapat diakses lewat sistem intranet perusahaan, sehingga dapat diakses dengan mudah oleh karyawan Dynapack Asia.

B. Saran

Adapun beberapa saran untuk penelitian selanjutnya:

1. Penambahan grafik interaktif pada bagian *reporting* agar pengguna lebih mudah untuk menangkap maksud dari laporan tersebut.
2. Migrasi dari dokumen lampiran hasil tes/*interview* menjadi *form* di website yang dapat diakses kapanpun, agar analisa data lebih mudah dilakukan

DAFTAR PUSTAKA

- [1] T. Yuniarsih and M. D. Sugiharto, "Human Resource Management Model to Create Superior Performance," *International Journal of Education*, pp. 74-81, 2016.
- [2] A. Halim, F.N. Ferdinand, C. S. Ko (2018), Ontology-Based Decision Support System for Hypersensitivity Disorder Allergy, *ICIC Express Letters*, vol.12, no.8, pp.847-854
- [3] DIJKSTRA Algorithm Based Approach to Shortest Path Model in Public Bus Transportation, *International Journal of Computer Science and Engineering (IJCSE)*, Vol 7 No.6., pp 1-8
- [4] B. Smith, *Advanced ActionScript 3*, New York: Springer, 2015.
- [5] Bello and R. Oluwaseun, "A University Examination Web Application Based on Linear-Sequential Life Cycle Model," *DIU Journal of Science and Technology*, vol. 12, no. 1, 2017.
- [6] M. F. F. A. Nasution and H. R. Weistroffer, "Documentation in Systems Development: A Significant Criterion for Project Success," *Hawaii International Conference on System Sciences*, 2009.
- [7] M. B. Romney and P. J. Steinbart, *Accounting Information Systems*, Global Edition, Harlow: Pearson Education Limited, 2014.