

Pengembangan Sistem Informasi Sumber Daya Manusia pada PT. Toyogiri Iron Steel berbasis Web

Mas Haq Kresno D. A. C¹, Denny Sagita Rusdianto², Eriq Muhammad Adams Jonemaro³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹mashaq.kresno@student.ub.ac.id, ²denny.sagita@ub.ac.id, ³eriq.adams@ub.ac.id

Abstrak

Terdapat masalah-masalah pada bagian HRD di PT.Toyogiri Iron Steel. Masalah yang pertama yaitu prosedur pengajuan lembur, cuti dan izin karyawan. Karyawan yang melakukan pengajuan harus meminta tanda tangan atasan dan pimpinan departemen sebelum disetujui HRD. Ini menimbulkan masalah, atasan tidak ada dalam pabrik atau tidak masuk. HRD juga belum bisa melakukan *monitoring* lembur karyawan. HRD mengecek karyawan yang melakukan lembur apakah sudah sesuai dengan jam yang diajukan. Belum terintegrasinya perhitungan gaji dengan presensi. Integrasi komponen seperti rekapitulasi data presensi, lembur, izin, cuti, tunjangan dan potongan dalam gaji. Bagian HRD memasukkan secara manual dimana waktu kurang efisien. Dan fungsi tambahan seperti pemberian surat peringatan, skorsing dan pengajuan promosi/demosi. Dari permasalahan tersebut, penulis akan membuat sistem informasi sumber daya manusia pada PT. Toyogiri Iron Steel berbasis web. Sistem ini berfungsi menyimpan informasi, membantu berjalannya pengajuan lembur, cuti dan izin, integrasi presensi dan penggajian, pengajuan promosi/demosi dan surat peringatan. Pengembangan sistem berbasis web yang dikembangkan dengan SDLC *waterfall style*. Sistem dibuat dengan perancangan berorientasi objek dan menggunakan metode MVC (*Model-View-Controller*) pada *framework CodeIgniter* dengan membagi menjadi 3 bagian, *model* untuk interaksi ke *database*, *controller* untuk logika utama, dan *view* untuk tampilan sistem, dan MySQL dan MariaDB sebagai *database*. Dari hasil rekayasa kebutuhan, terdapat 64 kebutuhan fungsional dan 1 non fungsional. Pada hasil perancangan dan implementasi, digunakan bahasa pemrograman PHP dengan *framework CodeIgniter*. Implementasi antarmuka menggunakan bahasa pemrograman HTML, CSS, dan *Javascript*. Pada pengujian sistem dilakukan pengujian unit dengan menggunakan *white box testing* dengan metode *basis path testing* diuji pada 3 sampel fungsi dengan semua bernilai *valid*. Pengujian validasi diuji ke semua kebutuhan fungsional dengan semua bernilai *valid*. Pengujian validasi *compatibility*, aplikasi sistem diuji di 6 *web browser* berbeda dengan hasil semua berjalan dengan baik.

Kata kunci: sumber daya manusia, pengajuan, presensi, perhitungan gaji, web, CodeIgniter

Abstract

There are problems in the HR department at PT.Toyogiri Iron Steel. The first problem is the procedure for applying for overtime, leave and permission. Employees who make submissions must ask for signatures from their superiors and department heads before being approved by HRD. This creates a problem, the boss is not in the factory or not present. HRD also cannot monitor employee overtime. HRD checks employees who are doing overtime whether it is in accordance with the proposed hours. Not yet integrated between salary calculation and attendance. The integration of components such as attendance data recapitulation, overtime, permits, leave, allowances and deductions in salary. The HRD entered manually where time was less efficient. And provides additional functions such as issuing warning letters, suspension and promotion / demotion. From these problems, the authors will create a web-based human resource information system at PT. Toyogiri Iron Steel. This system functions to store information, assist applications for overtime, leave and permits, integration of attendance and payroll, submit promotion / demotion and warnings. Development of a web-based system developed with the waterfall style SDLC. The system is created with object-oriented design and uses the MVC (Model-View-Controller) method in the CodeIgniter framework by dividing it into 3 parts, a model for interaction to the database, a controller for the main logic, and a view for the system view, and MySQL and MariaDB as the database. From the results of requirements engineering, there are 64 functional

requirements and 1 non-functional requirement. In the design and implementation results, the PHP programming language is used with the CodeIgniter framework. The interface implementation uses the HTML, CSS, and Javascript programming languages. In system testing, unit testing is carried out using white box testing with the basis path testing method tested on 3 sample functions with all valid values. Validation testing is tested for all functional requirements with all valid values. Validation testing for compatibility, system applications were tested on 6 different web browsers with all the results running well.

Keywords: human resources, submissions, attendance, salary calculation, web, CodeIgniter

1. PENDAHULUAN

Penggunaan teknologi informasi sudah diterapkan di segala bidang pekerjaan manusia. Penggunaan teknologi informasi diterapkan secara luas dalam bidang teknologi, ilmu terapan, seni budaya, pemerintahan, pendidikan, perdagangan, jurnalisme, dan lain-lain. Penggunaan teknologi informasi menggantikan sistem yang dahulu menggunakan kertas dan cara memprosesnya yang masih manual dengan dioperasikan oleh manusia. Komputerisasi menggantikan peran kertas dan pemrosesan informasi secara manual. Penggunaan teknologi informasi dalam suatu perusahaan digunakan dalam bidang administrasi perusahaan seperti pada bagian sumber daya manusia. Bagian HRD perusahaan mengatur seluruh data karyawan mulai dari penyimpanan data karyawan, posisi, presensi, pengajuan-pengajuan karyawan dan penggajian karyawan. PT. Toyogiri Iron Steel adalah sebuah perusahaan baja di Bekasi, Indonesia yang telah berdiri sejak 1974.

Terdapat masalah-masalah terkait aktivitas kerja dalam perusahaan berdasarkan hasil wawancara dengan bagian HRD PT. Toyogiri Iron Steel. Masalah yang pertama yaitu ada pada prosedur pengajuan lembur, cuti dan izin karyawan. Karyawan yang ingin melakukan pengajuan harus meminta tanda tangan atasan langsung dan pimpinan departemen (untuk cuti) sebelum disetujui oleh HRD. Hal ini menimbulkan masalah karena atasan tidak selalu ada dalam pabrik atau mungkin tidak masuk karena lain hal. Masalah selanjutnya yaitu HRD belum bisa melakukan *monitoring* lembur karyawan. HRD mengecek karyawan yang melakukan lembur apakah sudah bekerja sesuai jam yang diajukan atau belum. Masalah selanjutnya yaitu belum terintegrasinya penggajian dengan presensi karyawan. Integrasi komponen lainnya seperti rekapitulasi data presensi karyawan, lembur, izin, cuti, tunjangan dan potongan dalam gaji. Bagian HRD harus

memasukkan secara manual dimana kurang efisien secara waktu. Fungsi-fungsi tambahan untuk keperluan HRD lainnya seperti pemberian surat peringatan dan skorsing karyawan, jenjang karir karyawan, promosi dan demosi karyawan juga dibutuhkan.

Dengan sistem yang dipakai secara manual sekarang, bagian HRD banyak mengalami kendala dan proses yang panjang. Dalam melakukan presensi karyawan, bagian HRD mencatat jam masuk dan jam pulang dalam buku presensi. Untuk kebutuhan pengajuan karyawan, perusahaan menyediakan formulir sesuai kebutuhan. Karyawan dapat mengisi formulir cuti/izin/lembur dan diberi ke HRD untuk diproses. Untuk perhitungan gaji, dilakukan dengan menggunakan aplikasi *Microsoft Excel*. Bagian HRD mengisi *sheet* pada aplikasi secara manual dengan memasukkan data presensi dari buku presensi. Bagian HRD juga memasukkan pengajuan-pengajuan karyawan seperti cuti, izin dan lembur untuk diakumulasi pada perhitungan gaji karyawan. Proses secara panjang dan manual ini menimbulkan masalah seperti adanya kesalahan proses memasukkan data karena *human error* dan memakan waktu yang banyak untuk memasukkan data satu persatu dimana bagian HRD hanya mempunyai 3 karyawan.

Maka dari permasalahan di atas, penulis membuat solusi dengan merancang sistem yang dapat menyelesaikan permasalahan di atas. Penulis akan membuat sistem informasi sumber daya manusia pada PT. Toyogiri Iron Steel berbasis web. Sistem ini berfungsi menyimpan informasi karyawan, membantu berjalannya prosedur pengajuan lembur, cuti dan izin, integrasi presensi karyawan pada penggajian karyawan, pengajuan promosi/demosi dan surat peringatan. Pengembangan sistem berbasis web yang dikembangkan menggunakan SDLC *waterfall style*. Sistem dibuat dengan perancangan berorientasi objek dan menggunakan metode MVC (*Model-View-Controller*) pada *framework CodeIgniter*, dan

MySQL dan MariaDB sebagai *database* nya.

Software Development Life Cycle atau SDLC merupakan teknik dilakukan dalam pengembangan, implementasi, dan mengubah suatu sistem perangkat lunak dengan menggunakan metodologi tertentu (A.S & Shalahudin, 2018). Dalam pengembangan SDLC terdapat salah satu model yang sering digunakan yaitu *waterfall model* disebut model sekuensial linier atau alur hidup klasik (A.S & Shalahudin, 2018). Kemudian dalam perancangan sistem dapat digunakan *Unified Modeling Language* (UML). UML adalah bahasa spesifikasi yang menggambarkan, mendesain dan memvisualisasikan perancangan dalam pembuatan aplikasi, bagian dari informasi yang digunakan untuk dihasilkan dengan proses pembuatan perangkat lunak dan dapat berupa model atau deskripsi (Pratama, 2019). Sistem dibuat dengan berorientasi objek dengan menggunakan *CodeIgniter*. *CodeIgniter* merupakan *framework* pengembangan aplikasi digunakan untuk membangun aplikasi dengan simpel dan mudah sebagai alat untuk membuat bahasa pemrograman PHP mudah untuk dipakai (Upton, 2007). *CodeIgniter* mengandung konsep MVC (*Model-View-Controller*). MVC adalah pola pengembangan yang membagi kode menjadi 3 bagian utama, yaitu model untuk interaksi ke *database*, *controller* untuk logika utama, dan *view* untuk tampilan aplikasi.

Dari sistem yang dibuat, sistem diharapkan dapat memudahkan karyawan perusahaan PT. Toyogiri Iron Steel. Karyawan dapat dengan cepat dan mudah mengajukan pengajuan lembur, cuti dan izin. Dari sisi HRD, dapat mempermudah memproses pengajuan karyawan, presensi dan perhitungan gaji serta mengolah dan mencari data karyawan yang diperlukan sehingga lebih efektif dan efisien.

2. LANDASAN KEPUSTAKAAN

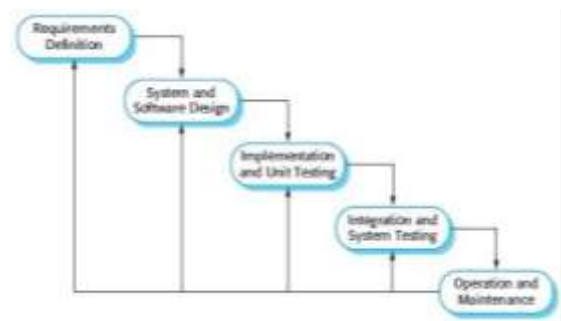
2.1 Tinjauan Pustaka

Pada penelitian yang berjudul “Pengembangan Sistem Informasi Keuangan Pegawai Biro Organisasi Sekretariat Daerah Provinsi Jawa Timur”, penelitian tersebut menyelesaikan masalah berupa sering terlambatnya dalam pemantauan dan pemberian tunjangan, perhitungan keuangan, tidak adanya pencatatan presensi karyawan, dan tidak ada pencatatan perjalanan dinas keluar kota. Maka dari itu, penulis mengembangkan sistem

informasi dengan menggunakan pendekatan orientasi objek dengan *framework codeigniter* dan bahasa pemrograman PHP, HTML, serta *javascript* (Mukarromah, et al., 2018).

2.2 Waterfall Model

System development life cycle (SDLC) adalah teknik dilakukan dalam pengembangan, implementasi, dan mengubah suatu sistem perangkat lunak dengan menggunakan metodologi tertentu (A.S & Shalahudin, 2018). Salah satu model dalam *system development life cycle* yang ada yaitu *waterfall model*. Model *waterfall* yaitu aktivitas dasar atas proses-proses yang tersusun atas spesifikasi, pengembangan, validasi, evolusi dan direpresentasikan ke dalam tahapan aktivitas pengembangan terpisah yang mendasar (Sommerville, 2011). Tahapan-tahapan pada *waterfall model* yaitu:



Gambar 1 *Waterfall Model*
 Sumber: (Sommerville, 2011)

2.3 Rekayasa Kebutuhan

Rekayasa Kebutuhan atau *requirement engineering* memiliki arti yaitu rangkaian atau urutan yang luas dari tugas-tugas dan teknik-teknik yang mengarah kepada pemahaman untuk suatu persyaratan. Dari sisi pembangunan perangkat lunak, rekayasa kebutuhan merupakan aksi utama dalam rekayasa perangkat lunak dan berlanjut ke pemodelan. Rekayasa kebutuhan berfungsi untuk membangun jembatan ke desain dan pembangunan perangkat lunak. Proses rekayasa dimulai dari *stakeholder* yang dimana fungsi bisnis didefinisikan, kemudian membangun skenario pengguna, dijabarkan dan dijelaskan fungsi-fungsi yang harus ada (Pressman & Maxim, 2015).

2.4 CodeIgniter

CodeIgniter merupakan *framework* pengembangan aplikasi digunakan untuk membangun aplikasi dengan simpel dan mudah sebagai alat untuk membuat bahasa

pemrograman PHP mudah untuk dipakai (Upton, 2007). Dengan memakai *CodeIgniter*, dapat menghemat waktu dengan memangkas kode yang diketik dan *script* dapat lebih mudah dibaca dan diperbarui dengan penggunaan struktur yang koheren (Upton, 2007). *CodeIgniter* mengandung konsep MVC (*Model-View-Controller*). MVC adalah pola pengembangan yang membagi kode menjadi 3 bagian utama, yaitu model untuk interaksi ke *database*, *controller* untuk logika utama, dan *view* untuk tampilan aplikasi.

3. METODOLOGI PENELITIAN

Tahapan pada metodologi penelitian mengikuti alur siklus *waterfall model*. Berikut adalah diagram alir dari metodologi penelitian:



Gambar 2 Diagram Alir Tahapan Penelitian

Pada studi literatur, peneliti mempelajari bahan-bahan literatur sesuai dengan apa yang ingin dibuat yaitu rancang bangun sistem informasi sumber daya manusia pada PT. Toyogiri Iron Steel berbasis web.

Pada rekayasa kebutuhan, peneliti melakukan penggalan kebutuhan terhadap sistem yang akan dibuat. Untuk itu peneliti melakukan tahapan proses yang terdiri dari elisitasi kebutuhan melalui wawancara kepada narasumber yaitu HRD dan analisis kebutuhan ditentukan aktor sistem, kebutuhan fungsional, spesifikasi kebutuhan yang menghasilkan pemodelan *use case diagram*.

Pada perancangan sistem, penulis merancang sistem dari hasil rekayasa kebutuhan. Tahapan-tahapannya yaitu perancangan

arsitektur, perancangan komponen, perancangan basis data dan perancangan antarmuka menghasilkan desain antarmuka. Pada implementasi sistem, penulis melakukan implementasi hasil rancangan sistem. Tahapan-tahapannya yaitu spesifikasi sistem, implementasi kode, implementasi basis data, dan implementasi antarmuka.

Pengujian sistem untuk memeriksa sistem yang telah diimplementasikan terhadap kebutuhan fungsional dan kebutuhan non-fungsional. Tahapan-tahapan yang dilakukan pada pengujian sistem yaitu Pengujian unit menggunakan *whitebox testing* dengan *basis path testing*. Pengujian validasi pada aksi-aksi yang bisa dilihat (*visible*) oleh aktor dan *output* yang dikenali oleh aktor untuk memvalidasi *output* dengan kasus uji didasarkan pada kebutuhan fungsional. Pengujian validasi *compatibility* dengan menggunakan berbagai macam perangkat lunak *web browser* yang tersedia. Pada kesimpulan, penulis melakukan pengambilan kesimpulan dari rekayasa kebutuhan, perancangan sistem, implementasi sistem, dan pengujian sistem. Pada saran, penulis menjabarkan tulisan berupa saran yang digunakan untuk pengembangan sistem lebih lanjut.

4. REKAYASA KEBUTUHAN

4.1 Elisitasi Kebutuhan

Elisitasi kebutuhan dilakukan untuk menemukan kebutuhan-kebutuhan yang ingin dicapai oleh para pengguna sistem dan dapat menjadi acuan dalam pendefinisian kebutuhan sistem yang akan dibuat. Proses bisnis berisi penjelasan alur pelaksanaan prosedur pada PT. Toyogiri Iron Steel dari awal hingga akhir. Proses elisitasi dilakukan oleh penulis melalui wawancara kepada narasumber yang mempunyai jabatan sebagai kepala HRD. Proses elisitasi dilakukan juga dengan observasi *Standard of Procedure* yang berlaku. Tugas HRD yaitu proses pengajuan lembur, pengajuan izin, pengajuan cuti, presensi, perhitungan gaji, promosi/demosi dan pemberian surat peringatan.

4.2 Analisis Kebutuhan

Analisis kebutuhan melakukan identifikasi aktor pada sistem yang dijabarkan pada Tabel 1.

Tabel 1 Identifikasi Aktor

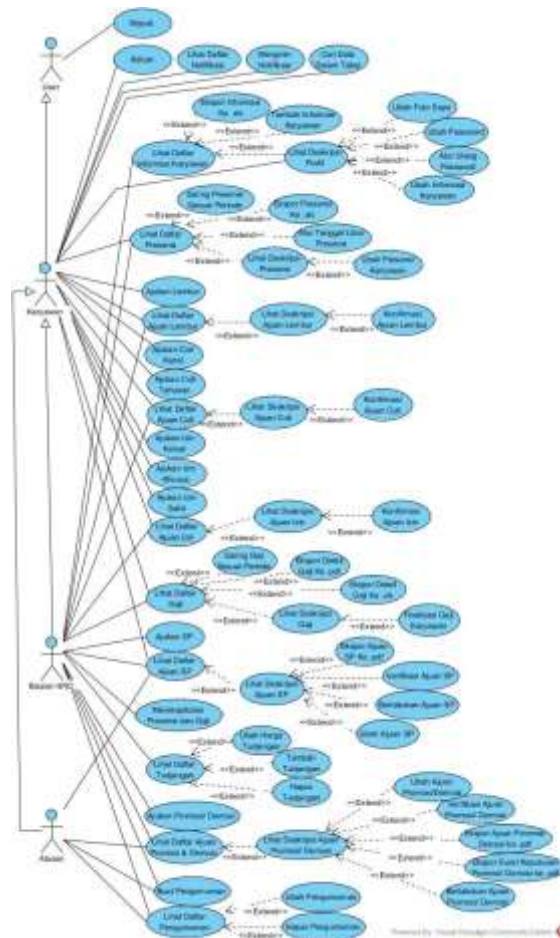
| No | Aktor | Deskripsi |
|----|------------|--|
| 1 | User | User sebagai pengguna sistem sebelum dapat masuk ke aplikasi. |
| 2 | Karyawan | Karyawan sebagai pengguna sistem untuk melakukan pengajuan-pengajuan dan sebagai pengguna sistem utama. |
| 3 | Bagian HRD | Bagian HRD sebagai aktor karyawan dan sebagai pengguna sistem yang memiliki fungsional tertentu seperti menyetujui pengajuan-pengajuan karyawan. |
| 4 | Atasan | Atasan sebagai aktor karyawan dan sebagai pengguna sistem yang sama seperti karyawan dengan memiliki fungsional khusus dalam sistem. |

Pada sistem, terdapat 64 kebutuhan fungsional dan 1 non fungsional. Contoh kebutuhan fungsional dapat dilihat pada Tabel 2.

Tabel 2 Kebutuhan Fungsional

| Kode Fungsi | Nama Fungsi | Deskripsi |
|-------------|------------------------------|---|
| TOY_1_16 | Ajukan Lembur | Sistem dapat memproses ajuan lembur dengan mengisi formulir. |
| TOY_1_17 | Lihat Daftar Ajuan Lembur | Sistem dapat menampilkan daftar ajuan lembur pengguna. |
| TOY_1_18 | Lihat Deskripsi Ajuan Lembur | Sistem dapat menampilkan deskripsi ajuan lembur pengguna. |
| TOY_1_19 | Ajukan Izin Keluar | Sistem dapat memproses ajuan izin keluar dengan mengisi formulir. |

Kemudian ditentukan pemodelan kebutuhan dengan digambarkan *use case diagram* yang menggambarkan kebutuhan fungsional pada Gambar 3.



Gambar 3 Use case diagram

5. PERANCANGAN DAN IMPLEMENTASI

5.1 Perancangan Arsitektur

Pada perancangan arsitektur dibuat pemodelan *sequence diagram* dan *class diagram*. Contoh salah satu *sequence diagram* fungsi Merkapitulasi Presensi dan Gaji pada Gambar 4. Untuk pemodelan *class diagram*, didapatkan 10 *class*. Terdiri 3 *class controller* dan 5 *class model*. *Class Diagram* sistem dapat dilihat pada Gambar 5.

5.2 Perancangan Komponen

Perancangan komponen dilakukan sebelum diimplementasikan ke dalam bahasa pemrograman, penulis membuat perancangan komponen sistem dengan membuat *pseudocode* dari fungsi sistem.

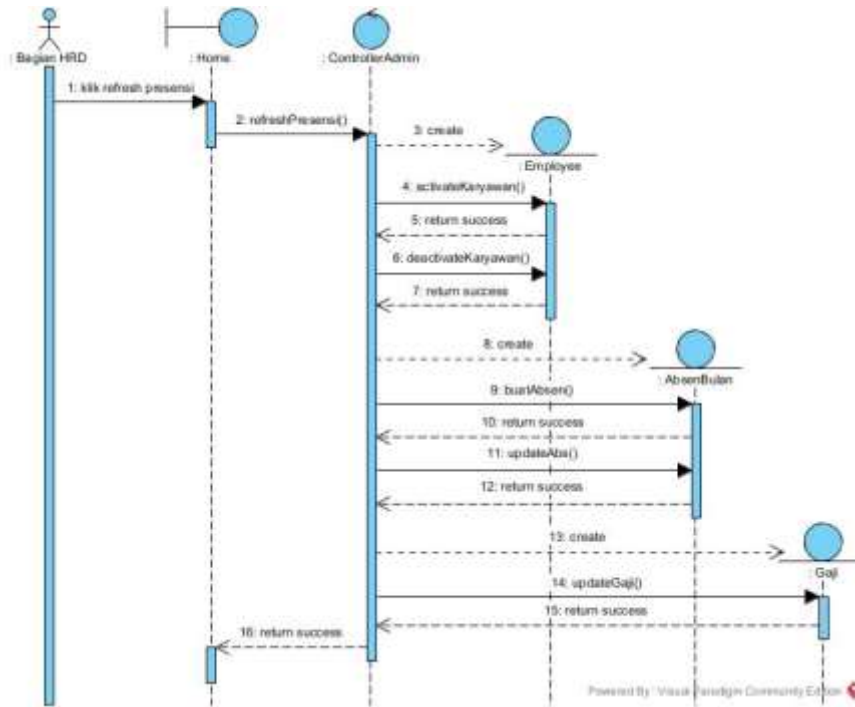
5.3 Perancangan Basis Data

Perancangan basis data sistem informasi sumber daya manusia pada PT. Toyogiri Iron Steel berbasis web menggunakan *Conceptual*

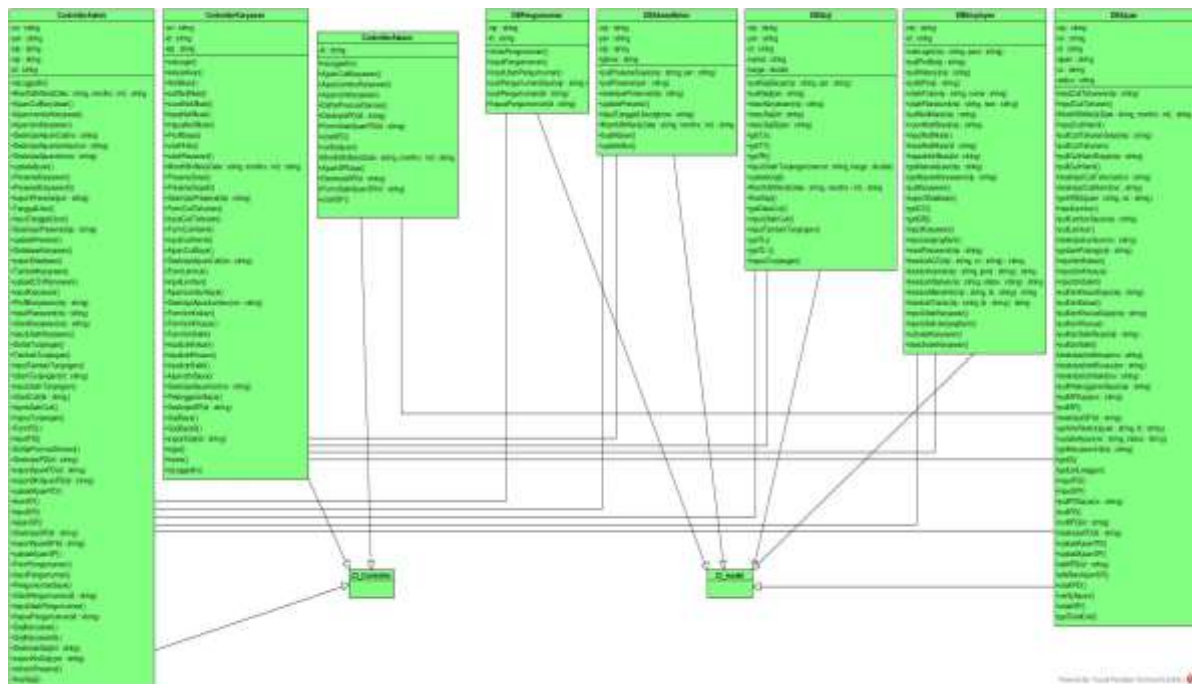
Data Model (CDM). Pada CDM terdapat 22 tabel data yang saling berelasi. Tabel yang ada yaitu tabel tunjangan_kantor, cost_center, shift, p_lembur, p_cutitahunan, presensi, notifikasi, historykarir, gaji, pengumuman, grade dan employee dan lain-lain.

5.4 Perancangan Antarmuka

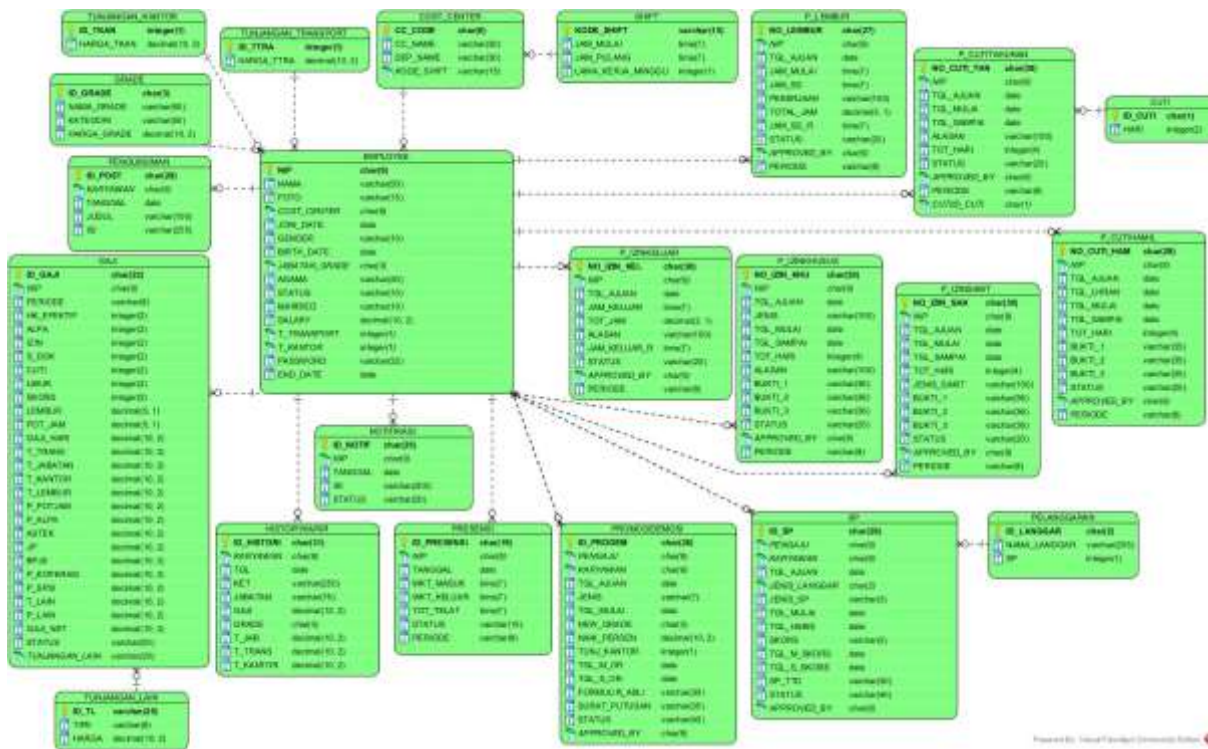
Perancangan antarmuka sistem informasi sumber daya manusia pada PT. Toyogiri Iron Steel berbasis web menggunakan *high-level design* sebelum diimplementasikan.



Gambar 4 Sequence Diagram Merkapitulasi Presensi dan Gaji



Gambar 5 Class Diagram Sistem



Gambar 6 Physical Data Model Sistem

5.5 Spesifikasi Sistem

Spesifikasi sistem menjelaskan spesifikasi yang dibutuhkan pada tahap implementasi sistem. Yang termasuk pada spesifikasi sistem yaitu perangkat keras dan perangkat lunak yang digunakan pada implementasi sistem.

5.6 Implementasi Kode Program

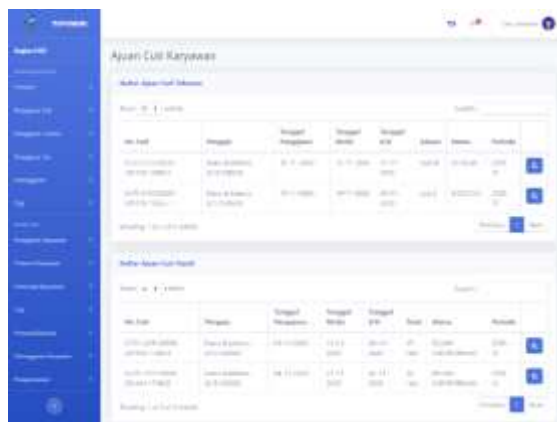
Implementasi kode program dengan rancangan pseudocode yang telah dibuat pada bab sebelumnya. Kode program ditulis dalam bahasa pemrograman PHP dengan menggunakan framework codeigniter yang terdiri atas model, view dan controller sistem.

5.7 Implementasi Basis Data

Implementasi basis data dilakukan implementasi dengan menggunakan physical data model yang sudah mewakili basis data sistem keseluruhan. Implementasi basis data dapat dilihat pada Gambar 6.

5.8 Implementasi Antarmuka

Implementasi antarmuka sistem dilakukan menggunakan bahasa pemrograman HTML, CSS, Javascript dan menggunakan Bootstrap, datatables dan fontawesome. Contoh implementasi antarmuka sistem pada Gambar 7.



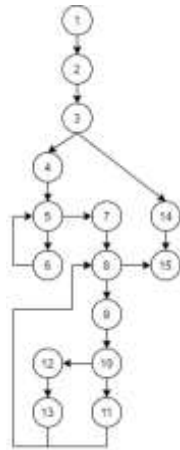
Gambar 7 Hasil implementasi antarmuka halaman daftar ajuan cuti

6. PENGUJIAN

6.1 Pengujian Unit

Pada pengujian unit, peneliti menggunakan White Box Testing dengan metode basis-path testing. Berikut adalah salah satu contoh pengujian unit yaitu fungsi updateAbs() pada merekapitulasi presensi dan gaji:

- a) Flow graph



Gambar 8 Flow graph dari fungsi updateAbs()

b) Cyclomatic Complexity (V(G))

1. $V(G) = \text{Total region} = 5$
2. $V(G) = \text{Total edge} - \text{Total node} + 2 = 18 - 15 + 2 = 5$
3. $V(G) = \text{Total predicate} + 1 = 4 + 1 = 5$

c) Independent Path

- Path 1 = 1-2-3-14-15
- Path 2 = 1-2-3-4-5-7-8-15
- Path 3 = 1-2-3-4-5-6-7-8-15
- Path 4 = 1-2-3-4-5-6-7-8-9-10-11-15
- Path 5 = 1-2-3-4-5-6-7-8-9-10-12-13-15

Kasus uji:

Tabel 3 Kasus Uji dan Hasil updateAbs()

| No. | Prosedur | Expected Result | Result | Status |
|-----|---|---|---|--------|
| 1 | File = tidak ada | Proses penarikan data fingerprint lewat | Proses penarikan data fingerprint lewat | Valid |
| 2 | File = "att20201120.txt" Isi = kosong | Proses penarikan data fingerprint tidak ada pembaruan presensi | Proses penarikan data fingerprint tidak ada pembaruan presensi | Valid |
| 3 | File = "att20201120.txt" Isi = ada | Proses penarikan data dengan adanya isi maka melakukan presensi | Proses penarikan data dengan adanya isi maka melakukan presensi | Valid |
| 4 | File = "att20201120.txt" Isi = ada Status = 1 | Proses penarikan data fingerprint | Proses penarikan data fingerprint | Valid |

| | | | | |
|---|---|--|--|-------|
| | | pembaruan presensi pada jam masuk | pembaruan presensi pada jam masuk | |
| 5 | File = "att20201120.txt" Isi = ada Status = 2 | Proses penarikan data fingerprint pembaruan presensi pada jam keluar | Proses penarikan data fingerprint pembaruan presensi pada jam keluar | Valid |

6.2 Pengujian Validasi

Pada pengujian validasi, akan dilakukan serangkaian pengujian pada kebutuhan fungsional sistem. Semua fungsionalitas sistem akan diuji dan diamati hasil eksekusi dari fungsinya berdasarkan pada use case scenario sistem. Dari hasil pengujian, didapatkan hasil uji 100% valid.

6.3 Pengujian Compatibility

Pada pengujian compatibility, memiliki fungsi untuk melihat apakah sistem dapat bekerja secara baik pada berbagai jenis web browser pada desktop. Pengujian compatibility diuji menggunakan aplikasi yaitu SortSite. SortSite adalah alat pengujian kualitas untuk web dengan berbagai parameter. Rincian pengujian ada pada Gambar 9.



Gambar 9 Hasil Pengujian Compatibility

Dari gambar di atas, dapat dilihat hasil pengujian sistem pada semua web browser dengan Critical Issues tidak ada. Ini berarti tidak ada konten atau fungsi yang hilang. Namun, terdapat beberapa poin masalah layout/performance pada Major Issues dan Minor Issues pada web browser Internet Explorer, Firefox, Safari, Opera, dan iOS.

7. KESIMPULAN DAN SARAN

7.1 Kesimpulan

Dari hasil analisis dan perancangan sistem informasi sumber daya manusia pada PT. Toyogiri Iron Steel berbasis web, terdapat aktor-aktor yang dapat menggunakan sistem yaitu

user, karyawan, bagian HRD, dan atasan. User sebagai pengguna sistem sebelum dapat masuk ke aplikasi, karyawan sebagai pengguna sistem untuk melakukan pengajuan dan sebagai pengguna sistem utama, bagian HRD sebagai karyawan dan pengguna sistem yang memiliki fungsional tertentu seperti menyetujui pengajuan karyawan dan lainnya, atasan sebagai karyawan dan pengguna sistem dengan memiliki fungsional tertentu seperti verifikasi ajuan SP dan lainnya. Dalam sistem, terdapat 64 kebutuhan fungsional dan 1 non fungsional. Kebutuhan fungsional mencakup pengajuan cuti, pengajuan izin, pengajuan lembur, presensi, perhitungan gaji, pengajuan promosi/demosi, pengajuan Surat Peringatan. Dalam perancangan sistem, dibuat perancangan arsitektur, perancangan komponen, perancangan basis data dan antarmuka. Perancangan arsitektur terdiri *sequence diagram* ajukan lembur, merekapitulasi presensi dan gaji, berlakukan ajuan surat peringatan dan *class diagram* terdiri 10 kelas. Perancangan komponen yaitu *pseudocode* dari sistem. Perancangan basis data terdiri *conceptual data model* (CDM) sistem terdapat 22 tabel data. Perancangan antarmuka terdiri atas *high-level design* dari tampilan sistem.

Pada proses implementasi sistem, terdapat implementasi kode program, implementasi basis data dan antarmuka. Implementasi kode digunakan bahasa pemrograman PHP dengan *framework CodeIgniter* dengan aplikasi text editor notepad++. Implementasi basis data menghasilkan *physical data model* (PDM) yang sudah mewakili basis data sistem keseluruhan menggunakan *phpmyadmin MySQL* dan *mariaDB*. Implementasi antarmuka menggunakan bahasa pemrograman HTML, CSS, *Javascript*, dengan *library Bootstrap*, *datatables*, *pdfviewer* dengan aplikasi *text editor* notepad++.

Pada pengujian sistem dilakukan pengujian unit, pengujian validasi, dan pengujian *compatibility*. Pengujian unit diuji pada 3 sampel fungsi pada sistem dengan hasil semua pengujian bernilai valid. Pengujian validasi diuji ke semua kebutuhan fungsional sistem dan hasil semua pengujian bernilai valid. Pada pengujian *compatibility*, aplikasi sistem diuji di berbagai *web browser* menggunakan aplikasi SortSite. Setelah dijalankan, tidak ada *critical issues* namun ditemukan beberapa *major* dan *minor layout issues* pada *web browser* Internet Explorer, Firefox dan Safari.

7.2 Saran

Saran untuk individu yang ingin mengembangkan sistem atau melanjutkan sistem seperti penulis telah buat, yaitu sebagai berikut:

1. Penambahan fitur pengajuan mutasi departemen untuk karyawan dengan atasan dapat mengajukan ajuan mutasi kepada karyawan untuk pindah ke departemen lainnya.
2. Fitur penambahan, pengurangan dan perubahan *shift* kerja karyawan seperti hari kerja, jam masuk dan jam keluar karyawan untuk dapat diberlakukan secara dinamis kepada setiap karyawan sesuai kebutuhan perusahaan.

8. DAFTAR PUSTAKA

- A.S, R. & Shalahudin, M., 2018. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika.
- Mukarromah, L., Pradana, F. & Saputra, M., 2018. Pengembangan Sistem Informasi Keuangan Pegawai Biro Organisasi Sekretariat Daerah Provinsi Jawa Timur. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, Volume III, pp. 2043-2052.
- Pratama, A. R., 2019. *Unified Modeling Language*. [Online] Available at: <https://www.codepolitan.com/unified-modeling-language-uml> [Accessed 1 September 2019].
- Pressman, R. S. & Maxim, B. R., 2015. *Software Engineering Practitioner's Approach*. 8th ed. New York: Mc Graw Hill.
- Sommerville, I., 2011. *Software Engineering*. 9th ed. Boston: Addison-Wesley.
- Upton, D., 2007. *Codeigniter for Rapid Php Application Development*. Birmingham: Packt Publishing.