

## Pengembangan Sistem Informasi Penggajian Pada Tugu Hotel Lombok Menggunakan Metode *Rational Unified Process*

Anas Zulbichaq<sup>1</sup>, Yusi Tyroni Musrsityo<sup>2</sup>, Mochamad Chandra Saputra<sup>3</sup>

Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Brawijaya  
Email: <sup>1</sup>zulbichaqanas@gmail.com, <sup>2</sup>yusi.tyro@gmail.com, <sup>3</sup>andra@ub.ac.id

### Abstrak

Kalkulasi penggajian pegawai pada Tugu Hotel Lombok masih dinilai kurang efisien dan efektif. Peningkatan efisiensi dan efektifitas dapat dicapai dengan adanya sistem informasi yang membantu proses kalkulasi. Penelitian pengembangan sistem informasi penggajian dilakukan guna menghasilkan purwarupa sistem informasi yang dapat meningkatkan efisiensi dan menekan angka human error pada proses kalkulasi gaji pegawai. Analisis masalah dilakukan dengan wawancara dengan metode tidak terstruktur. Setelah dilakukan aktifitas analisis masalah dan studi literatur, dilakukan proses pengembangan sistem informasi menggunakan metode RUP (*Rational Unified Process*). Fase inepsi penelitian ini dilakukan untuk mendefinisikan proses bisnis as-is dan proses bisnis to-be, identifikasi kebutuhan fungsional dan non-fungsional, dan identifikasi fitur sistem. Fase elaborasi menghasilkan diagram UML berupa use case diagram yang dijadikan acuan untuk memodelkan diagram perancangan seperti activity diagram dan sequence diagram. Hasil pemodelan diagram kelas analisis dijadikan acuan pada perancangan physical data model. Aktifitas analisis masalah dan perancangan sistem ini dilakukan dengan pendekatan berbasis objek (OOA/D). Fase konstruksi dilakukan aktifitas pengembangan sistem dilakukan menggunakan pendekatan OOP. Setelah melalui fase konstruksi, akan dilakukan pengujian terhadap sistem yang telah dibangun. Pengujian dilakukan menggunakan metode database testing, validation testing, dan compatibility testing. Pengujian database testing dilakukan pada fase konstruksi. Pada fase transisi berupa pengujian validation testing dan compatibility testing. Pengujian pada fase transisi dilakukan pada fitur-fitur utama pada system, dan pengujian kompatibility dilakukan guna mengetahui tingkat kehandalan system pada lingkungan yang berbeda. Hasil pengujian menyatakan bahwa sistem informasi dapat berjalan sesuai spesifikasi dan kebutuhan sehingga dapat digunakan oleh pengguna.

**Kata kunci:** *sistem informasi, sistem penggajian, RUP, UML, pengujian sistem*

### Abstract

The employee payroll calculation of Tugu Hotel Lombok is still less efficient and effective. Efficiency and effectiveness improvement could be achieved with presence an information system which can help out calculation process. Research on development of payroll information system is to produce an prototype information system that could give efficiency improvement and reduce human error on employee payroll calculation. Res analysis done using unstructure interview method. After did res analysis and literature study, do development process use RUP method. Inception phase will be modeled the as-is business process and to-be business process, feature, functional and non-functional requirement. In elaboration phase deliver UML diagram like use case diagram as a reference to modeling some diagram else like activity diagram and sequence diagram. The result of analysis class diagram as reference to physical data modeling. Analysis and designing using object oriented method (OOA/D). in construction phase is developing system using OOP method approach. After develop the system according by design before, next step is carried out test the system. Testing use database testing, validation testing, and compatibility testing methods. Database testing done on construction phase. In transition phase is done compatibility testing and validation testing. Database testing and validation testing is done on main features of this system, and compatibility testing is done to find out

feasibility system in different environment. The test result claim that information system run according by requirements and could used easily by the users.

**Keywords:** *information system, payroll system, RUP, UML, system testing*

---

## 1. PENDAHULUAN

Hotel Tugu Lombok adalah perusahaan yang bergerak dalam bidang perhotelan. Pada struktur organisasi yang bersifat *back office* terdapat suatu jabatan yang bernama manajer operasional. Manajer operasional adalah jabatan yang bertugas yang untuk menjamin kesejahteraan karyawan pada Tugu Hotel Lombok. salah satu tugas dari manajer operasional adalah menghitung gaji beserta bonus tambahan untuk setiap karyawan. Menurut *restaurant head department*, pada setiap bulannya manajer operasional harus melakukan kalkulasi yang bersifat semi otomatis. Dalam praktiknya manajer operasional Tugu Hotel Lombok masih mengandalkan *Microsoft Office (Microsoft excel)* untuk melakukan kalkulasinya. Biasanya manajer menghabiskan waktu sekitar 2 jam untuk memastikan hasil kalkulasinya sudah benar. Salah satu penyebab yang membuat pekerjaan tersebut memakan waktu cukup lama adalah manajer harus menginputkan secara manual data penilaian subyektif dari seluruh karyawan pada setiap divisi.

Karena hanya mengandalkan perangkat lunak tersebut manajer sering mengalami masalah dalam melakukan penghitungan, terlebih apabila terjadi kesalahan pada saat menginput data. Hal ini berpotensi terjadinya kerugian ketika nominal yang di hasilkan tidak valid. Selain itu, masalah yang nampak adalah adanya penilaian harian yang ditulis tangan pada secarik kertas. Hal tersebut sangat rawan dengan tindak kelalaian yang berpotensi merugikan sebelah pihak. Terlebih lagi apabila pada akhir bulan setiap *head department* harus memberikan total poin harian yang telah dicatat selama satu bulan pada form penilaian dan data complain pegawai. Tidak jarang terdapat kelalaian pada saat penjumlahan total nilai harian pegawai pada proses tersebut.

Berdasarkan masalah tersebut terdapat sebuah solusi dengan adanya sistem yang melakukan proses penghitungan gaji dan bonus. Nantinya sistem ini akan melakukan proses penghitungan gaji sesuai dengan standar Tugu Hotel Lombok. Penerapan sistem informasi

secara efektif mampu menyajikan informasi yang akurat dan dapat dipercaya, dan juga akan meningkatkan efisiensi kerja pegawai (Astuti and Dharmadiaksa 2014). Penggunaan sistem ini diharapkan dapat membantu manajer Tugu Hotel Lombok dalam menghitung dan menyimpan data akan gaji dan bonus karyawannya. Karena penyimpanan data menggunakan *database* sehingga data yang tersimpan akan aman dari adanya bencana alam. Pada proses pengembangan sistem ini terdapat sebuah metode pendekatan yang dinilai sesuai dengan studi kasus masalah penggajian pada Tugu Hotel Lombok ini, metode pendekatan sistem informasi menggunakan pendekatan metode Rational Unified Process (RUP). RUP merupakan suatu siklus pengembangan rekayasa perangkat lunak bersifat iteratif yang dikembangkan dengan mengumpulkan *best practices* yang terdapat dalam pengembangan perangkat lunak (Anwar 2014). Iteratif disini merupakan proses perulangan pada fase yang terdapat pada RUP, fase RUP yang dimaksud diantaranya fase *inception, elaboration, construction* dan *transition*. Perulangan pada setiap fase mengakomodasi adanya perubahan kebutuhan perangkat lunak. Pengembangan sistem informasi penggajian Tugu Hotel Lombok ini juga akan dilakukan secara bertahap dan memungkinkan adanya perubahan kebutuhan dan penambahan fitur. Sistem informasi akan dikembangkan menggunakan Bahasa PHP dengan menggunakan pendekatan OOP. Pengujian sistem dilakukan dengan metode pendekatan *database testing, validation testing, dan compatibility testing*.

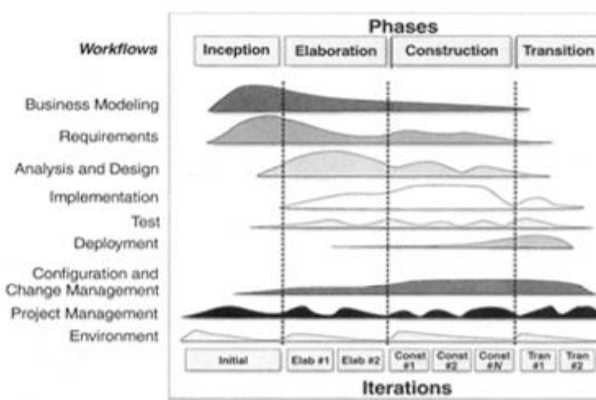
## 2. LANDASAN TEORI

### 2.1. Rational Unified Process

*Rational Unified Process* merupakan sebuah metode pendekatan proses pengembangan yang bersifat iteratif. Untuk sebuah sistem sederhana, RUP sangat layak untuk mendefinisikan setiap masalah, merancang solusi, membangun perangkat lunak, dan menguji produk yang dihasilkan

secara bertahap. Pendekatan secara iteratif mampu meningkatkan pemahaman terkait masalah melalui penyempurnaan berulang dan berkembang secara bertahap pada solusi efektif selama beberapa siklus.

Membangun dengan pendekatan iteratif merupakan sebuah fleksibilitas untuk mengakomodasi kebutuhan-kebutuhan baru atau perubahan-perubahan taktik dalam tujuan bisnis. Selain itu juga memungkinkan proyek untuk diidentifikasi dan mengatasi risiko yang ada lebih cepat (Booch, Rumbaugh and Jacobson 1999). Setiap fase yang ada pada RUP dapat dilakukan iterasi. Iterasi merupakan sebuah putaran pengembangan atau pengulangan pengembangan pada produk yang dikerjakan sehingga nantinya akan menghasilkan sebuah produk akhir berupa sistem yang sudah *final* (Booch, Rumbaugh and Jacobson 1999).



Gambar 1 Fase RUP

Sumber: (Krutchen 2000)

Gambar 2.1 merupakan gambaran ilustrasi fase-fase dan aktifitas pada RUP. Pada metode RUP terdapat 4 fase yaitu insepisi, elaborasi, konstruksi dan transisi.

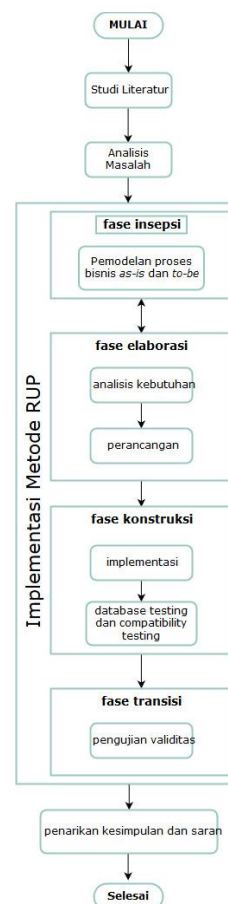
2.2. Pemodelan Berorientasi Objek

Pemodelan berorientasi objek merupakan metode-metode yang digunakan untuk melakukan analisis dan desain untuk sistem yang menggunakan metode pengembangan berorientasi objek. Terdapat beberapa metode yang berorientasi objek antara lain Object Oriented Analysis (OOA), Object Oriented Design (OOD), dan Object Oriented Programming (OOP). OOA merupakan metode pendekatan untuk menganalisis spesifikasi dan kebutuhan sistem yang akan akan dibangun

menggunakan konsep OOP (Rosa & Shalahuddin, 2014).

Hasil dari OOA ini dapat berupa diagram *use case* dan *class diagram* pada UML. OOD merupakan sebuah metode pendekatan untuk melakukan perancangan sistem berdasarkan hasil analisis pada OOA (Rosa & Shalahuddin, 2014). Secara sederhana OOA merupakan aktifitas untuk mengidentifikasi masalah dan memberikan *deliverable* berupa diagram *use case* dan *class diagram* pada UML. Sedangkan OOD merupakan aktifitas perancangan sistem yang akan dikembangkan, biasanya *deliverable* pada aktifitas ini berupa diagram alur dan sketsa atarmuka pengguna sistem. OOP merupakan kensep pemrograman yang dilakukan berdasarkan hasil OOA/D yang telah dilakukan.

3. METODOLOGI



Gambar 2 metodologi penelitian

Gambar 2 merupakan alur pada penelitian ini. Alur dimulai dengan kegiatan studi literatur, kemudian dilanjutkan dengan analisis masalah. setelah itu dilanjutkan dengan implementasi fase-fase pada metode RUP, dan diakhiri oleh penarikan kesimpulan.

### 3.1 Fase Insepsi

Pada fase ini terdapat beberapa aktifitas yang dilakukan yang pertama adalah mendefinisikan ulang proses bisnis *as-is*. Proses bisnis *as-is* ini didapatkan dari hasil wawancara bersama responden. Tujuan mendefinisikan ulang proses bisnis saat ini adalah untuk memudahkan peneliti dalam menganalisis masalah dan dapat memberikan solusi atas masalah yang terdapat pada proses bisnis tersebut. Solusi berdasarkan masalah pada proses bisnis saat ini biasanya disebut dengan proses bisnis *to-be*. Proses bisnis *to-be* yang dimodelkan adalah proses bisnis perbaikan berdasarkan adanya permasalahan pada proses bisnis *as-is*. Fokus proses bisnis yang dianalisa pada fase insepsi ini adalah proses bisnis penggajian pada Tugu Hotel Lombok. Luaran yang diharapkan fase ini adalah sebuah dokumen berupa perbandingan proses bisnis *as-is* dengan proses bisnis *to-be*. Dokumen perbandingan tersebut nantinya akan digunakan sebagai *deliverable* pada fase ini dan juga sebagai dokumen untuk melakukan diskusi terkait solusi proses bisnis baru yang diajukan pada pihak *stakeholder*

### 3.2 Fase Elaborasi

. Pada fase ini juga dilakukan analisis kebutuhan dan evaluasi yang memungkinkan adanya perubahan kebutuhan apabila ditemukan kebutuhan maupun membutuhkan penambahan fitur. Analisis kebutuhan merupakan aktifitas pertama yang dilakukan pada fase elaborasi ini. Pada tahap ini dilakukan identifikasi kebutuhan fungsional dan non-fungsional. Dokumen identifikasi kebutuhan tersebut nantinya akan dijadikan acuan untuk perancangan sistem. Pada tahap ini juga terdapat beberapa aktifitas perancangan diantaranya merepresentasikan fitur-fitur pada sistem yang akan dikembangkan, melalui diagram ERD sebagai skema basis data dan UML yang meliputi *use case diagram*, *activity diagram*, *sequence diagram*, dan *class diagram*.

### 3.3 Fase Konstruksi

Fase ini merupakan aktifitas implementasi hasil perancangan pada fase sebelumnya dan melakukan pengujian terhadap sistem. Pada fase ini terdapat dua aktifitas yaitu koding dan pengujian sistem yang telah dibangun.

Implementasi adalah tahap dimana pengembang melakukan koding sesuai dengan perancangan yang sudah dilakukan pada fase sebelumnya. Sistem ini nantinya akan dibangun menggunakan Bahasa pemrograman PHP dengan menggunakan kerangka kerja CodeIgniter dan MySQL sebagai DBMS yang digunakan untuk mengolah data-data penggajian Tugu Hotel Lombok.

Menurut (Sommerville, 2011) pengujian merupakan sesuatu yang telah prediksi untuk menunjukkan apa saja yang program dapat lakukan. Pengujian ini dilakukan guna untuk menemukan masalah lebih dini dan dilakukannya perbaikan sebelum sistem ini diperkenalkan pada pihak *stakeholder*. Metode pengujian yang dilakukan pada fase ini adalah *database testing*.

### 3.4 Fase Transisi

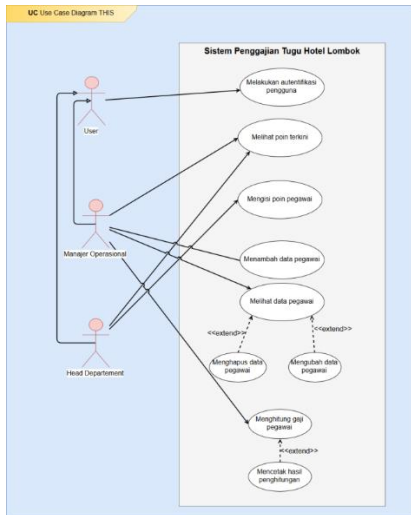
Transisi yang dilakukan pada fase ini adalah dengan cara memperkenalkan sistem pada pihak *stakeholder* dengan menggunakan hasil dokumentasi analisis kebutuhan, dokumentasi perancangan, dan dokumentasi pengujian sistem ini. Dokumen pengujian yang disertakan pada fase transisi ini adalah pengujian validasi dan kompatibilitas.

## 4. Hasil dan Pembahasan

### 4.1 Analisis Pesyarat

Setelah melakukan wawancara dengan tujuan mengumpulkan data-data, langkah selanjutnya adalah identifikasi masalah. Identifikasi masalah direpresentasikan dengan sebuah diagram proses bisnis penggajian.

Berdasarkan hasil wawancara didapatkan aktifitas yang menyebabkan proses bisnis tersebut kurang efisien. Setelah mengetahui permasalahan yang terjadi dilakukan pemodelan proses bisnis *to-be*. Terdapat dua proses bisnis yang diajukan sebagai solusi permasalahan tersebut. Kemudian dilakukan identifikasi kebutuhan fungsional dan non-fungsional dan dilanjutkan dengan membuat perancangan model *use case* untuk sistem. Hasil pemodelan *use case* terdapat pada gambar 2.



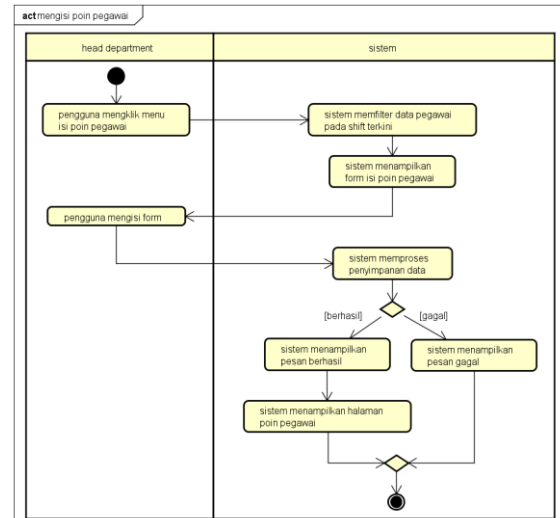
Gambar 3 diagram use case

pada diagram use case diatas terdapat 2 aktor yang dapat berkomunikasi dengan sistem dan terdapat 8 use case yang dapat membantu meringankan pekerjaan proses penggajian ini. Masing-masing use case memiliki kondisi persyaratan dan scenario dalam menjalankan use case tersebut. tabel 1 merupakan salah satu contoh scenario pada use case mengisi poin pegawai.

Tabel 1 skenario use case mengisi poin pegawai

<b>Brief Description</b>	Use case ini menggambarkan bagaimana langkah-langkah untuk mengisi poin pegawai harian.
<b>Actor</b>	head department
<b>Pre-condition</b>	<ol style="list-style-type: none"> <li>1. Komputer terhubung dengan server.</li> <li>2. Pengguna berhasil melakukan proses autentifikasi pengguna terhadap sistem.</li> <li>3. Pengguna dikenali system sebagai head department.</li> </ol>
<b>Post-condition</b>	<ol style="list-style-type: none"> <li>1. Sistem berhasil menyimpan poin pegawai.</li> </ol>
<b>Basic Flow</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu isi poin pegawai.</li> <li>2. Sistem memilah data pegawai pada shift terkini.</li> <li>3. Sistem menampilkan formulir penilaian pegawai.</li> <li>4. Aktor mengisi form.</li> <li>5. System memproses penyimpanan data</li> </ol>
<b>Alternative Flow</b>	

Selanjutnya adalah memodelkan scenario alur pada use case dalam bentuk activity diagram. Berdasarkan skenario use case yang telah dibuat sebelumnya, pada gambar 3 merupakan contoh activity diagram untuk mengisi poin pegawai.



Gambar 4 Activity Diagram isi poin pegawai

Pada fase ini terdapat 3 dokumen deliverable yang akan disampaikan pada stakeholder, dokumen tersebut antara lain diagram use case, use case spesifikasi, dan diagram aktifitas. Setelah dilakukan brainstorming untuk meningkatkan pemahaman stakeholder akan sistem yang akan dibangun, terdapat perubahan analisis persyaratan yang mengakibatkan terdapat penambahan use case dan fitur pada sistem ini. berikut use case yang akan digunakan sebagai acuan pada pengembangan sistem.

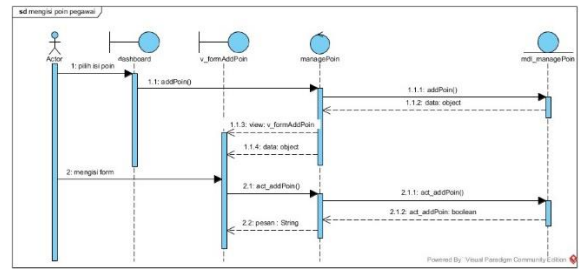


Gambar 5 perubahan use case

Gambar 5 merupakan diagram *use case* setelah dilakukan brainstorming dengan pihak stakeholder. Perubahan terjadi dikarenakan terdapat objek yang dapat mempengaruhi kalkulasi yaitu jumlah complain yang didapat pegawai. Sehingga terdapat fitur baru untuk mengolah data complain pegawai.

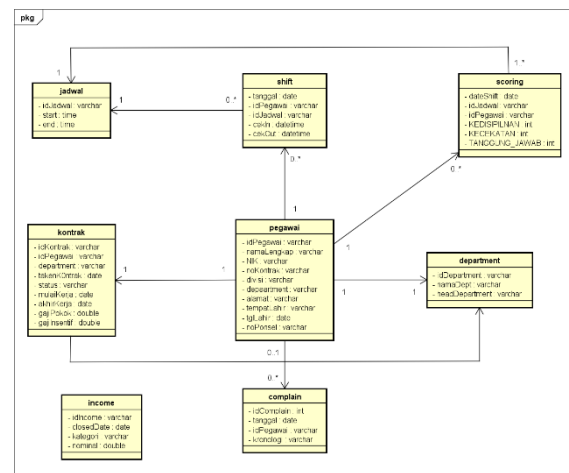
4.2 Perancangan

Bab perancangan sistem ini berisi tentang notasi pemodelan berdasarkan pemodelan UML pada bab sebelumnya. Notasi pemodelan sistem yang akan dijelaskan pada bab ini terdiri dari *sequence diagram*, *class diagram*, perancangan basis data yang dimodelkan dengan *physical data model* (PDM), dan perancangan antar muka pengguna. Hasil pemodelan *sequence diagram* terdapat pada gambar 6.



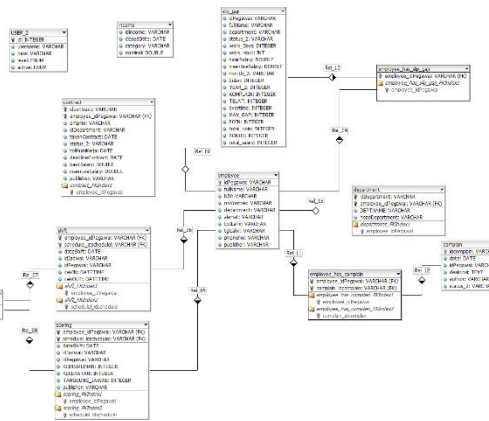
Gambar 6 sequence diagram isi poin pegawai

Diagram alur tersebut dimodelkan berdasarkan skenario alur yang telah dibuat pada use case spesifikasi. Gambar diatas merupakan gambaran objek-objek yang berkomunikasi untuk memberikan luaran sesuai pada perancangan use case. Objek yang terlibat adalah *managePoint* sebagai *controller* dan *mdl\_managePoin* sebagai *modeler*.



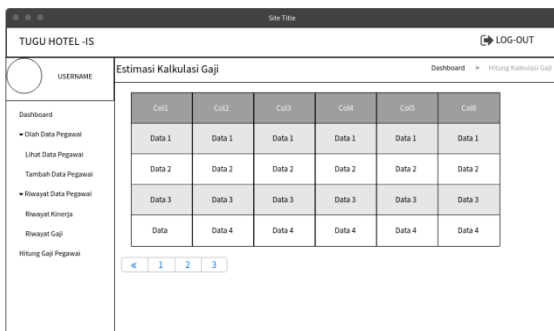
Gambar 7 diagram kelas analisis

Gambar 6 merupakan hasil pemodelan diagram kelas analisis. Diagram tersebut dimodelkan berdasarkan objek-objek apa yang terlibat pada proses penggajian. Diagram kelas analisis ini nantinya yang dijadikan acuan untuk memodelkan *physical data model* (PDM) untuk menampung data-data yang dibutuhkan. Skema PDM pada system ini terdapat pada gambar 8.



Gambar 8 skema table basis data

Selanjutnya adalah membuat sketsa antarmuka pengguna yang akan di implementasikan nantinya. Tujuan adanya perancangan antarmuka ini adalah sebagai acuan pada saat melakukan pengembangan antarmuka pengguna.



Gambar 9 sketsa antarmuka hitung gaji pegawai

Gambar 9 merupakan sketsa antarmuka pengguna hitung gaji pegawai. Pada sketsa halaman hitung gaji pegawai terdapat tabel yang berisikan data-data komponen kalkulasi penggajian. Tabel tersebut akan menampilkan informasi-informasi berupa identitas pegawai, kumulatif overtime pegawai, kumulatif menit terlambat seorang pegawai, total poin pegawai, dan total gaji yang didapat oleh pegawai.

### 4.3 Implementasi

Implementasi dilakukan dengan melakukan koding pada ahasa PHP dan menggunakan framework CI. Implementasi pada fungsi hitung gaji pegawai terdapat pada tabel 2.

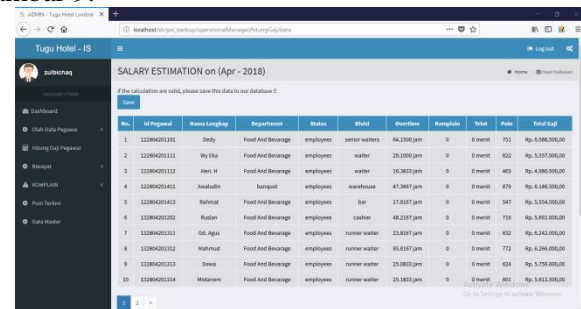
Tabel 2 kode program

Baris	Kode program
1	public function data(){
2	\$data['user'] = \$this->mdl_hitungGaji->data();

```

3   $this->load-
   >view('operationalManager/v_hitungGaji',
   $data);
4   }
    
```

Pada kode program diatas fungsi data akan mengakses model bernama hitungGaji. Pada penelitian ini kalkulasi gaji dilakukan menggunakan Bahasa SQL dan disimpan sementara pada tabel view DBMS. Hasil dari kalkulasi tersebut akan ditampilkan pada antarmuka pengguna yang terdapat pada gambar 9.



Gambar 10 hasil implmentasi antarmuka hitung gaji

Gambar 9 merupakan implmentasi sketsa antarmuka pengguna pada halaman hitung gaji pegawai yang telah dirancang sebelumnya. Pada halaman tersebut terdapat konten utama sebesar 9 kolom grid yang berisi informasi hasil kalkulasi.

## 5. PENGUJIAN

Pada penelitian ini dokumen test case mengadopsi dokumen test case yang disarankan oleh (B.B.Agarwal, et al., 2010) terdapat pada gambar 9.

Test case name	Test Case ID
Purpose of test	Testing object (unit, application, module, etc.)
Test attribute	
Tests focus (function, feature, process, interface, validation, verification, etc.)	
Test type (alpha, beta, unit, integration, system)	
Test process	A set of instructions for conducting the test-initial stating condition-inputs-specifications-output expected
Test results	Expected and actual and comparison, error description, post-process state
Action	Correction, authorization, and feedback through retest
Action to initialize the pre-test status	

Gambar 11 template dokumen test-case

Pengujian yang pertama dilakukan adalah menggunakan pendekatan metode database testing. Pada dasarnya pengujian basis data digunakan untuk menguji validitas data, integritas data, menguji performa relasi pada basis data, dan menguji UDF, Procedure, dan Trigger (Pressman, 2010). Pengujian dengan metode ini diuji pada fitur hitung gaji pegawai,

olah poin pegawai, dan olah data complain pegawai. Berdasarkan hasil pengujian yang dilakukan dengan menggunakan 12 *test case* bernilai valid pada sleuruh *test case*. Hal tersebut menandakan bahwa sintaks query yang diterapkan pada fitur-fitur utama sistem ini 100% berjalan dengan baik dan tidak terdapat masalah pada luaran sintaks query terkait.

Pengujian kedua adalah pengujian menggunakan metode pendekatan *validation testing*. Menurut (Pressman, 2010) pengujian validasi berdasarkan pada *usecase scenario* sehingga dapat diketahui kesalahan interaksi pada sistem. Kesimpulan yang didapat dari serangkaian pengujian validasi ini adalah sistem ini telah memenuhi seluruh kebutuhan pengguna. Pengujian ini dilakukan berdasarkan use case pada fungsi-fungsi yang berkaitan dengan fitur utama sistem. Hal tersebut berdasarkan tidak adanya hasil pengujian yang bernilai tidak valid, sehingga sistem ini layak untuk dilakukan penyempurnaan lebih lanjut dari purwa rupa sistem ini.

Pengujian terakhir merupakan pengujian dengan menggunakan metode pendekatan *compatibility testing*. Pengujian kompatibilitas ini merupakan metode yang ditujukan untuk mengetahui kemampuan sistem berbasis WebApp dapat digunakan dengan lancar pada lingkungan yang berbeda sebelum sistem ini benar-benar diluncurkan secara online (Pressman, 2010). Pada penelitian ini pengujian ini akan dibantu oleh sebuah *tools* bernama Sortsite.

Hasil uji pada pengujian ini bernilai valid karena sesuai dengan *expectation result* pada pengujian ini. Hasil pengujian ini berdasarkan laporan hasil uji pada *tools* sortsite ini.

Browser	IE	Edge	Firefox	Safari	Opera	Chrome	iOS	Android
Version	11	17	61	11	54	68	≤9 10 11	≤3 4*
Critical Issues	●	●	●	●	●	●	●	●
Major Issues	●	●	●	●	●	●	●	●
Minor Issues	●	●	●	●	●	●	●	●

Gambar 12 hasil pengujian kompatibilitas

Pada gambar 7.4 terdapat 3 *critical issue* pada pengujian ini. *Critical issue* pada *browser* firefox dan chrome karena terdapat halaman yang menggunakan *input type* berupa “password”. Untuk versi terakhir pada piranti peramban firefox dan chrome sangat disarankan menggunakan protokol HTTPS untuk menjaga keamanan kata sandi yang diketikan pada kolom isian halaman terkait. Hal tersebutlah yang membuat *tools* sortsite menganggap

masalah tersebut menjadi masalah yang sangat serius.

Priority 1

2 issues on 2 pages

- ▶ ● Chrome and Firefox display multiple security warnings for password fields that don't use HTTPS.
- ▶ ● The element.attachEvent method was removed in Internet Explorer 11.

Gambar 13 Critical Issue

Dikarenakan sistem ini masih berupa purwa-rupa dan tidak perlu untuk dilakukan instalasi pada suatu server, sehingga mengubah protokol sistem menjadi HTTPS. Pada gambar 7.5 merupakan masalah-masalah yang ditemukan melalui pengujian yang dilakuka *sortsite*. Selain terdapat *critical issue* pada piranti chrome dan firefox, juga terdapat *critical issue* pada piranti Internet Explorer. Pada piranti IE, *critical issue* berupa tidak berfungsinya suatu method javascript yang digunakan untuk efek animasi. Hal tersebut telah diinformasikan sebelumnya oleh pihak AdminLte selaku pengembang *template* antarmuka yang digunakan pada sistem ini yang menyatakan bahwa peramban IE tidak mendukung untuk efek transisi atau animasi.

6. KESIMPULAN dan SARAN

Kesimpulan dari penelitian penegmbangan sistem informasi penggajian dengan menggunakan metode RUP telah memberikan solusi dari masalah yang terjadi pada proses penggajian Tugu Hotel Lombok. Sistem yang dikembangkan pada penelitian ini menggunakan pendekatan berbasis objek, sehingga aktifitas analisis dan perancangan system juga berbasis objek. Analisis masalah dilakukan dengan kegiatan wawancara menggunakan metode semi-struktur kemudian dimodelkannya proses bisnis saat ini dalam diagram BPMN untuk meningkatkan pemahaman proses. Setelah itu dilakukan pemodelan proses bisnis *to-be* sebagai solusi untuk mengatasi masalah pada proses bisnis saat ini, terdapat dua proses bisnis *to-be* pada penelitian ini. Analisis proses bisnis tersebut berdasarkan objek-objek yang terlibat dalam proses bisnis tersebut.

Setelah kesepakatan proses bisnis *to-be*, dilakukan aktifitas identifikasi kebutuhan. Identifikasi kebutuhan ditujukan untuk mengetahui kebutuhan system dan indiviti yang berinteraksi dengan system nantinya.setelah itu dilakukan perancangan use-case dan diagram UML yang dijadikan acuan pada proses



pengembangan sistem. Terdapat 9 usecase pada sistem.

Sistem dikembangkan menggunakan kerangka kerja CodeIgniter, kerangka kerja tersebut mengimplementasikan pendekatan OOP dan berdasarkan perancangan yang telah dibuat sebelumnya. Hasil uji yang dilakukan terhadap system ini dilakukan menggunakan 3 metode, yaitu *database testing*, *validation testing*, dan *compatibility testing*. Hasil uji yang telah dilaksanakan adalah system dapat memberikan informasi hasil kalkulasi dengan akurat, dan system dapat berjalan sesuai dengan perancangan. Sehingga sistem yang telah dikembangkan dapat dikembangkan lebih lanjut dengan mengintegrasikan data cek-log pegawai dengan basis data sistem ini.

Sommerville, Ian. *Software Engineering, Ninth Edition*. Boston: Addison-Wesley, 2011.

Sugiyono. *Metode Penelitian Pendidikan Pendekatan Kuantitatif, Kualitatif, dan R&D*. Bandung: Alfabeta Bandung, 2013.

## DAFTAR PUSTAKA

- Anwar, A. "A Review of RUP (Rational Unified Process)." *International Journal of Software Engineering (IJSE)* 5, no. 2 (2014): 70.
- Astuti, Ni Made Marlita, and Ida Bagus Dharmadiaksa. "PENGARUH EFEKTIVITAS PENERAPAN SISTEM INFORMASI AKUNTANSI, PEMANFAATAN DAN KESESUAIAN TUGAS PADA KINERJA KARYAWAN." *E-Jurnal Akuntansi Universitas Udayana*, 2014: 373-384.
- B.B.Agarwal, S. P. Tayal, and M. Gupta. *Software Engineering & Testing*. USA: Jones and Bartlett Publishers, 2010.
- Bittner, Kurt, and Ian Spence. *Use Case Modelling*. 1. Boston: Addison Wesley, 2002.
- Booch, G, J Rumbaugh, and I Jacobson. *The Unified Modelling Language User Guide*. first. Massachusetts: Addison-Wesley Longman, Inc, 1999.
- Krutchen, Philippe. *The Rational Unified Process: An Introduction*. Second. Massachusetts: Addison-Wesley, 2000.
- Pressman, Roger S. *Software Engineering: A Particioner's Approach*. seventh. New York: The McGraw-Hill Companies, Inc, 2010.
- Rosa, A.S, and M Shalahuddin. *REkayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. 2. Bandung: Informatika Bandung, 2014.