

PERANCANGAN APLIKASI RENCANA ANGGARAN BIAYA (RAB) (STUDI KASUS PADA DINAS PEKERJAAN UMUM KOTA SALATIGA)

Adi Nugroho¹, Yos Richard Beeh², Hettyca Astuningdyas³

Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana

Jl. Diponegoro 52-60, Salatiga 50711, Indonesia

E-mail: cmbeling@gmail.com¹, yos.fti.uksw@gmail.com², astuningdyas@yahoo.com³

ABSTRAK: Pembuatan *Project Cost Estimation* (PCE), daftar kebutuhan material, dan tenaga kerja seharusnya dilakukan sebelum suatu proyek dimulai. Proses ini membutuhkan analisis yang akurat dan berhati-hati. Saat ini Dinas Pekerjaan Umum (DPU) Salatiga masih mengerjakannya secara manual, dimana hal ini membutuhkan waktu yang lama dan akurasi yang kurang terjamin. Makalah ini membahas tentang perancangan dan pembuatan perangkat lunak komputer PCE yang efisien dan akurat, terutama untuk pembangunan dan perbaikan gedung. Perangkat lunak aplikasi PCE akan dikembangkan menggunakan metoda prototipe. Aplikasi dikembangkan menggunakan bahasa pemrograman Java, *Object Relational Mapping* (ORM) Hibernate, server basis data MySQL.

Kata kunci: *project cost estimation*, konstruksi bangunan dan rehabilitasi, java, *hibernate*, *prototyping*

ABSTRACT: Creating *Project Cost Estimation* (PCE), list of materials and staff requirements of a construction project should be done before do the project itself. This process needs careful and accurate analysis. Until now Dinas Pekerjaan Umum (DPU) in Salatiga city still made PCE manually. It needs much of time and lack of accuracy. This article studies about how to design and implements efficient and accurate PCE computer software, especially for building construction and rehabilitation. PCE application software will be by using *prototyping* method. The application is implemented with Java programming language, *Object Relational Mapping* (ORM) Hibernate, database server MySQL.

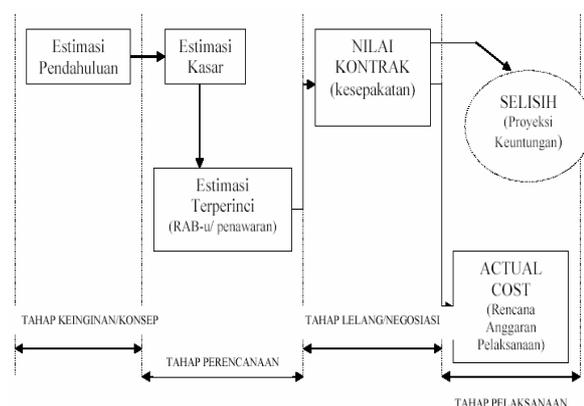
Keywords: *project cost estimation*, building construction and rehabilitation, java, *hibernate*, *prototyping*

PENDAHULUAN

Penyusunan Rencana Anggaran Biaya (RAB) suatu proyek adalah kegiatan yang harus dilakukan sebelum proyek dilaksanakan. RAB adalah banyaknya biaya yang dibutuhkan baik upah maupun bahan dalam sebuah pekerjaan proyek konstruksi. Daftar ini berisi volume, harga satuan, serta total harga dari berbagai macam jenis material dan upah tenaga yang dibutuhkan untuk pelaksanaan proyek tersebut.

Pada Dinas Pekerjaan Umum (DPU) kota Salatiga, RAB dipergunakan untuk merencanakan jumlah biaya yang dibutuhkan dalam pekerjaan baik gedung milik negara yang ditangani oleh bidang Cipta Karya, sarana pengairan yang ditangani oleh bidang Pengairan, serta jalan dan jembatan yang ditangani oleh bidang Bina Program. Penghitungan RAB didasarkan pada suatu analisis yang dituangkan dalam Peraturan Walikota (Perwali) Salatiga tentang standarisasi indeks biaya di lingkungan kota Salatiga. Perwali ini dibuat berdasarkan Standar Nasional Indonesia (SNI) dan *Burgeslijke Openbare Werken* (BOW). Saat ini Perwali terbaru yang mengatur tentang indeks biaya di lingkungan kota Salatiga ialah Perwali Salatiga No. 35 Tahun 2008 [1]. Saat ini pembuatan RAB sampai penjabarannya menjadi

daftar kebutuhan material dan upah tenaga di DPU kota Salatiga masih dilakukan secara manual, sehingga membutuhkan waktu yang cukup panjang dan seringkali terjadi *human error* yang menyebabkan perhitungan menjadi salah dan merugikan beberapa pihak yang terlibat.



Gambar 1. Tahapan dan Proses Penyusunan RAB

RENCANA ANGGARAN BIAYA

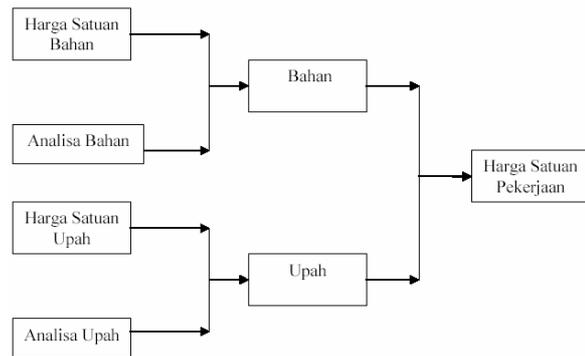
Rencana anggaran biaya proyek adalah perhitungan banyaknya anggaran biaya suatu bangunan

dan upah, serta biaya-biaya lain yang berhubungan dengan pelaksanaan proyek tersebut [2]. Definisi lain mengatakan RAB proyek adalah suatu proses perhitungan volume pekerjaan, harga dari berbagai macam bahan dan pekerjaan yang terjadi pada suatu konstruksi [3]. Dari kedua definisi di atas dapat disimpulkan bahwa RAB proyek adalah perhitungan banyaknya biaya yang diperlukan dalam suatu proyek konstruksi yang terdiri dari biaya bahan, upah tenaga, serta biaya lain yang berhubungan dengan proyek tersebut berdasarkan perhitungan volume pekerjaan yang telah dilakukan sebelumnya.

Berdasarkan Gambar 1, penyusunan RAB terbagi atas 2 bagian yaitu RAB terperinci dan RAB kasar. RAB kasar merupakan rencana anggaran biaya sementara dimana pekerjaan dihitung tiap ukuran luas. Pengalaman kerja sangat mempengaruhi penafsiran biaya secara kasar. Pada umumnya, hasil dari RAB kasar ini, apabila dibandingkan dengan RAB yang dihitung secara terperinci, akan terdapat selisih. Selain dari pengalaman, untuk proyek pemerintah biasanya sudah ditentukan pedoman harga satuan. RAB terperinci adalah anggaran biaya bangunan atau proyek yang dihitung dengan terperinci dan cermat, sesuai dengan ketentuan dan syarat-syarat penyusunan anggaran biaya. Adapun untuk proyek pemerintah biasanya telah ditetapkan daftar tingkat upah, bahan dan harga alat [3].

Penyusunan RAB secara terperinci pada dasarnya membutuhkan 5 hal yang paling mendasar, yaitu bestek dan gambar-gambar bestek, daftar upah, daftar harga bahan-bahan (material), daftar analisis, serta daftar volume tiap jenis pekerjaan yang ada. Daftar tersebut dapat saling memberikan gambaran dan petunjuk-petunjuk hingga akhirnya dapat merupakan anggaran biaya [4]. Di dalam RAB terdapat analisis harga satuan pekerjaan. Analisis harga satuan pekerjaan merupakan analisis bahan dan upah untuk membuat satu satuan pekerjaan tertentu, seperti 1 m^3 beton (1:2:3), 1 m^3 galian pondasi dan sebagainya, semuanya diatur dalam pasal-pasal pada buku BOW (*Burgelijke Openbare Werken*) maupun SNI (Standar Nasional Indonesia). Di Salatiga sendiri analisis harga satuan pekerjaan dituangkan dalam Perwali Salatiga tentang standarisasi indeks biaya di lingkungan pemerintah kota Salatiga. Adapun perwali yang terbaru adalah peraturan walikota Salatiga nomor 35 tahun 2008 tentang standarisasi indeks biaya di lingkungan pemerintah kota Salatiga [1].

Harga satuan pekerjaan terdiri atas tiga komponen, yaitu analisis harga satuan bahan/material, analisis harga satuan upah tenaga dan analisis harga satuan sewa alat yang bersifat opsional. Untuk lebih jelasnya dapat dilihat pada Gambar 2.

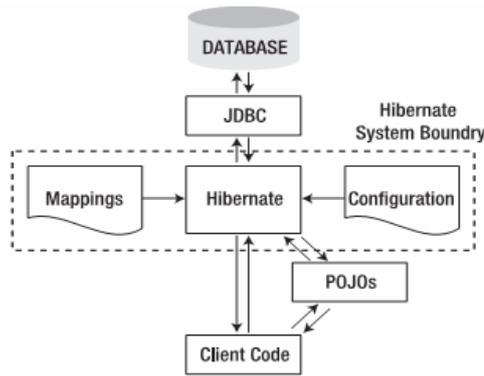


Gambar 2. Harga Satuan Pekerjaan

Proses analisis harga satuan bahan/material pada dasarnya adalah menghitung banyaknya volume masing-masing bahan serta besarnya biaya yang dibutuhkan untuk menyelesaikan per-satuan pekerjaan konstruksi. Analisis harga satuan bahan/material mengandung dua unsur yaitu: a) Harga satuan bahan, merupakan harga satuan bahan/material bangunan yang berlaku di pasar pada saat anggaran biaya bangunan tersebut disusun, dan b) Koefisien bahan, yaitu koefisien yang menunjukkan kebutuhan bahan/material bangunan untuk setiap satuan jenis pekerjaan. Proses analisis harga satuan upah tenaga pada dasarnya adalah menghitung banyaknya tenaga serta biaya yang dibutuhkan, untuk menyelesaikan per-satuan pekerjaan konstruksi. Analisis harga satuan upah tenaga mengandung dua unsur yaitu: a) Harga satuan upah tenaga, merupakan upah yang diberikan kepada tenaga kerja konstruksi perharinya atas jasa tenaga yang dilakukan sesuai dengan keterampilannya, dan b) Koefisien tenaga, yaitu koefisien yang menunjukkan kebutuhan tenaga kerja untuk tiap-tiap posisi. Sementara itu analisis harga satuan sewa alat pada dasarnya adalah menghitung banyaknya alat yang digunakan serta besarnya biaya sewa alat, untuk menyelesaikan per-satuan pekerjaan konstruksi. Analisis harga satuan sewa alat mengandung dua unsur, yaitu: a) Harga satuan sewa alat, merupakan harga satuan sewa alat yang berlaku di pasar pada saat anggaran biaya bangunan tersebut disusun, dan b) Koefisien alat, yaitu koefisien yang menunjukkan kebutuhan alat untuk setiap satuan jenis pekerjaan.

OBJECT RELATIONAL MAPPING

Object Relational Mapping (ORM) adalah sebuah teknik pemrograman untuk mengkonversi data antara sistem database relasional menjadi objek-objek dalam bahasa pemrograman yang bersifat *object-oriented*. ORM menciptakan sebuah objek database yang bersifat *virtual* yang dapat dikenali dan digunakan dalam bahasa pemrograman [5].



Gambar 3. Letak *Hibernate* dalam Aplikasi

HIBERNATE

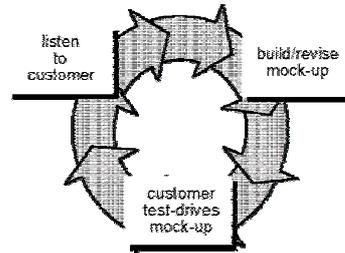
Hibernate adalah sebuah *tool* yang menjadi solusi untuk mengatasi masalah pengaturan data *persistent* dalam bahasa pemrograman Java. *Hibernate* menjadi pengatur interaksi antara aplikasi dengan basis data, sehingga pemrogram komputer dapat lebih berkonsentrasi mengembangkan logika bisnis aplikasi [5]. *Hibernate* adalah sebuah solusi terhadap masalah-masalah persistensi karena *Hibernate* memiliki hal-hal seperti tidak mengharuskan pemetaan satu POJO (*Plain Old Java Object*) menjadi satu tabel. Selain itu *Hibernate* mendukung berbagai jenis relasi antarkelas seperti *inheritance* dan yang lainnya dan telah diakui sebagai perangkat yang cepat walaupun ada beberapa performa yang membutuhkan waktu seperti saat *Hibernate* dijalankan dan melakukan pemrosesan file konfigurasi.

Cara kerja *Hibernate* adalah dengan menyediakan konfigurasi dan pemetaan antara aplikasi dengan basis data sehingga aplikasi dapat menyimpan POJO (*Plain Old Java Object*) dalam basis data. Pada Gambar 3 diperlihatkan letak *Hibernate* dalam aplikasi. Aplikasi dengan *Hibernate* akan menghasilkan baris-baris *coding* yang lebih sedikit dibandingkan dengan aplikasi yang pemetaannya dilakukan secara manual. Dengan pemrograman yang lebih sederhana, pemeliharaan aplikasi akan lebih mudah dilakukan [5].

METODE PROTOTYPING

Metode rekayasa perangkat lunak yang digunakan dalam pembuatan aplikasi pada artikel ini ialah metode *prototyping*. Metode *prototyping* merupakan sebuah metode rekayasa perangkat lunak yang bersifat iteratif. Metode ini menuntut adanya hubungan kerja yang dekat atau komunikasi intensif antara pembangun aplikasi dengan pengguna. Adapun keuntungan menggunakan metode *prototyping* adalah metode *prototyping* melibatkan partisipasi aktif dari pengguna sehingga dapat meningkatkan moral pengguna dalam

menggunakan aplikasi yang dihasilkan. Gambar 4 merupakan gambar tahapan yang harus dilakukan dalam metode *prototyping* [6].

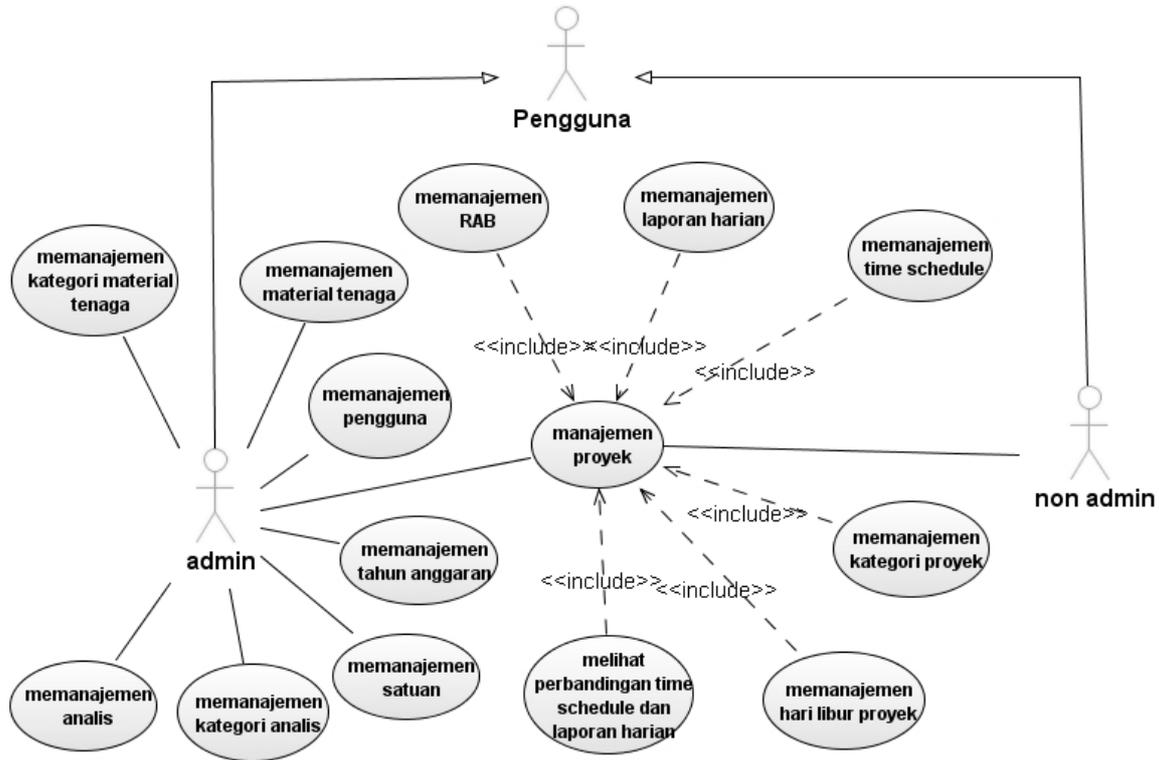


Gambar 4. Tahapan Metode Prototyping

Pada tahap *listen to customer* atau mengumpulkan informasi tentang kebutuhan aplikasi yang akan dibangun penulis melakukan tahap wawancara dengan pengguna staf bagian Cipta Karya. Setelah mengetahui kebutuhan umum aplikasi yang akan di bangun maka dilakukan studi pustaka tentang bagaimana cara membuat RAB, komponen-komponen yang ada dalam RAB, bagaimana mengurai RAB menjadi daftar kebutuhan material dan tenaga, bagaimana membuat *time schedule* serta laporan harian yang disesuaikan dengan proses yang ada di DPU Salatiga.

Tahapan selanjutnya dalam metode *prototyping* yaitu *build/revise mock-up* atau membangun aplikasi secara cepat. Pada tahap ini dilakukan pembuatan aplikasi secara cepat, lebih memfokuskan pada input-output aplikasi sesuai dengan kebutuhan umum yang diketahui pada tahap pertama. Tahap ini menghasilkan RAB *prototype 1*. Setelah RAB *prototype 1* dihasilkan proses memasuki tahapan selanjutnya yaitu *customer test-drives mock-up*. Pada tahap ini RAB *prototype 1* diserahkan kepada pengguna untuk di evaluasi oleh pengguna dan mendiskusikan solusi untuk kendala-kendala yang dialami pada saat pembuatan *prototype*.

Pada tahap penyerahan *prototype 1* didapatkan informasi baru tentang kebutuhan aplikasi yang dibangun nantinya. Setelah mendapatkan informasi baru tentang kebutuhan aplikasi, RAB *prototype 1* dikembangkan sesuai dengan kebutuhan baru hasil evaluasi *prototype 1* menjadi RAB *prototype 2*. Pembangunan RAB *prototype 2* ini juga lebih menekankan pada proses input dan output. Setelah RAB *prototype 2* selesai dibangun RAB *prototype 2* diserahkan kepada pengguna untuk di evaluasi oleh pengguna. Dari hasil evaluasi RAB *prototype 2*, pengguna menyatakan bahwa RAB *prototype 2* sudah memenuhi kebutuhan. Dengan demikian, proses berhenti karena tahapan dalam metode *prototyping* dinyatakan selesai jika pengguna menyatakan bahwa *prototype* yang dibangun sudah memenuhi kebutuhan pengguna.



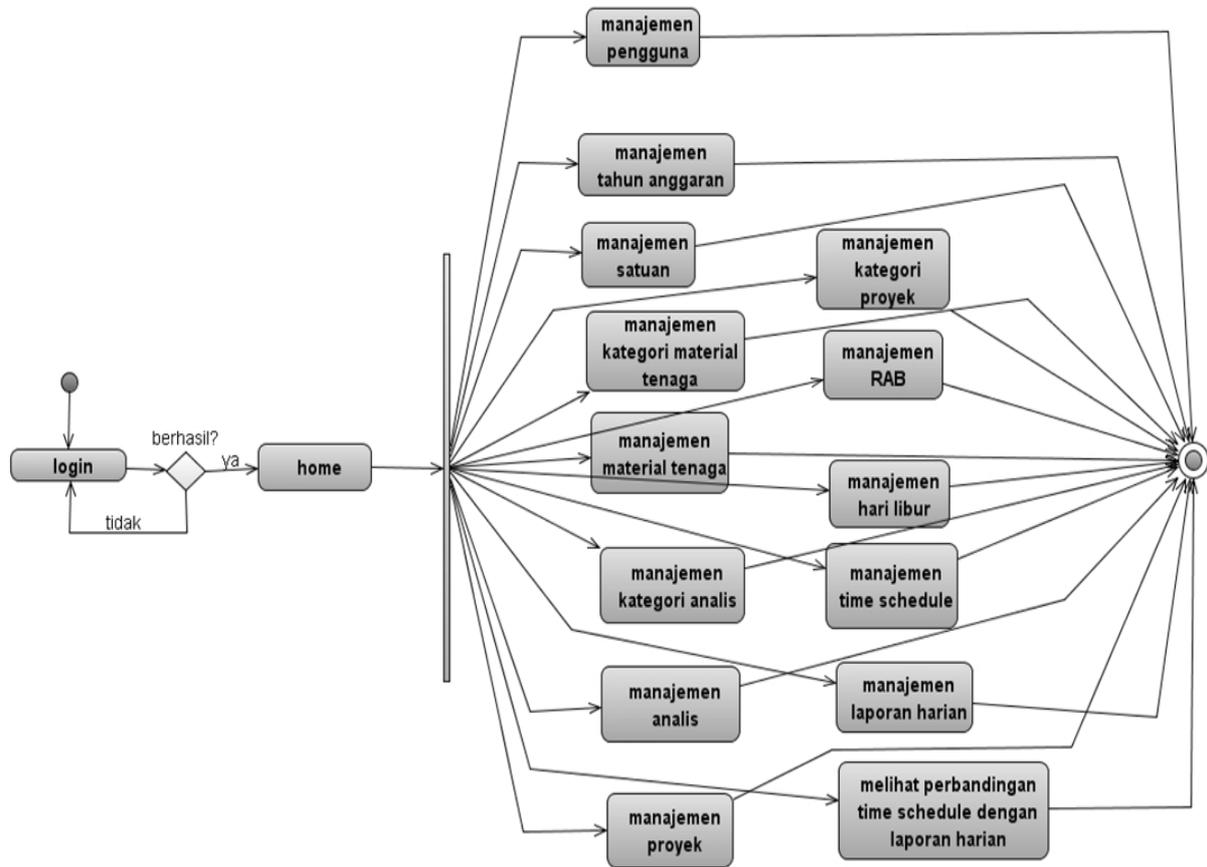
Gambar 5 Use Case Diagram Aplikasi RAB Gedung (ARuNG)

PERANCANGAN DAN IMPLEMENTASI APLIKASI RAB

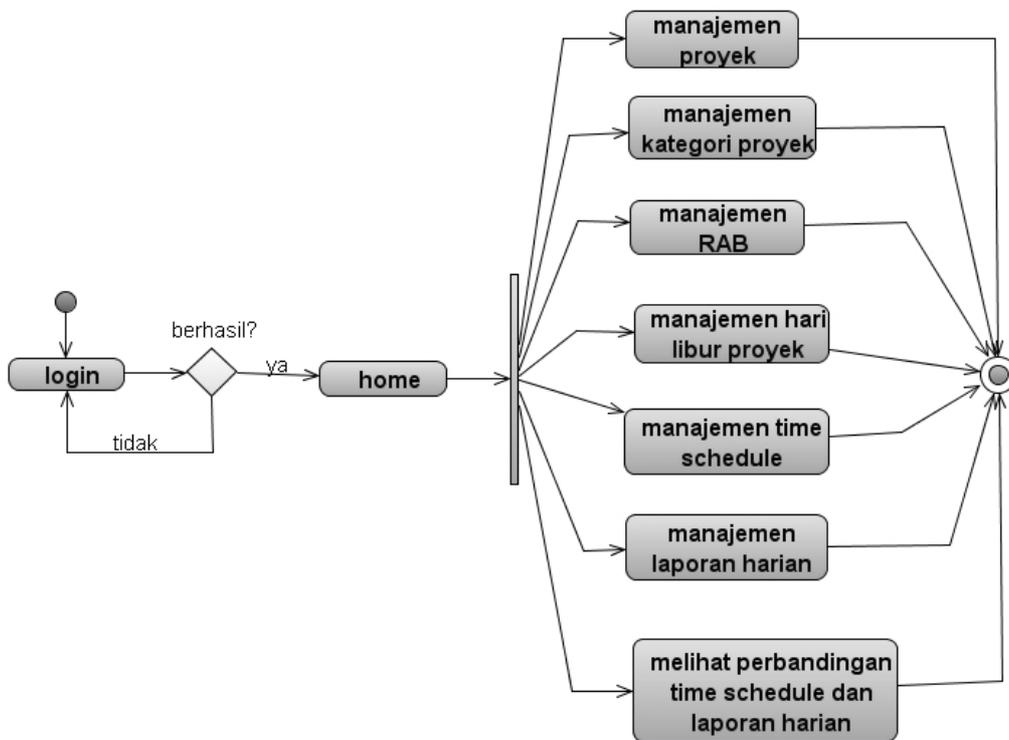
Dari hasil wawancara serta evaluasi yang didapatkan pada penyerahan setiap *prototype*, dapat diketahui fungsionalitas yang harus dimiliki oleh aplikasi yang dibangun. Hal ini digambarkan pada *use case diagram* yang ada pada Gambar 5. Dari *use case diagram* tersebut terdapat 2 jenis pengguna (*actor*) aplikasi yaitu admin dan non admin. Admin dan non admin merupakan generalisasi dari pengguna (*actor*) secara umum. Pengguna dalam sistem ini ialah pegawai dari Dinas Pekerjaan Umum Kota Salatiga. Admin ialah kepala DPU kota Salatiga, kepala bidang Bina Marga, serta kepala bidang Cipta Karya. Sedangkan yang termasuk non admin ialah staff bidang Bina Marga, dan staff bidang Cipta Karya.

Activity diagram untuk admin diperlihatkan dalam Gambar 6 dan digunakan untuk mendokumentasikan alur kerja pada sebuah sistem. Dalam aplikasi yang dibuat terdapat 2 aktor yang berinteraksi dengan sistem, yaitu admin dan non admin. Admin merupakan aktor yang dapat mengelola semua data di dalamnya, yaitu data master dan juga data proyek. Non admin merupakan aktor yang dapat mengelola data-data tertentu. Non admin mempunyai 7 aktivitas seperti diperlihatkan pada Gambar 7.

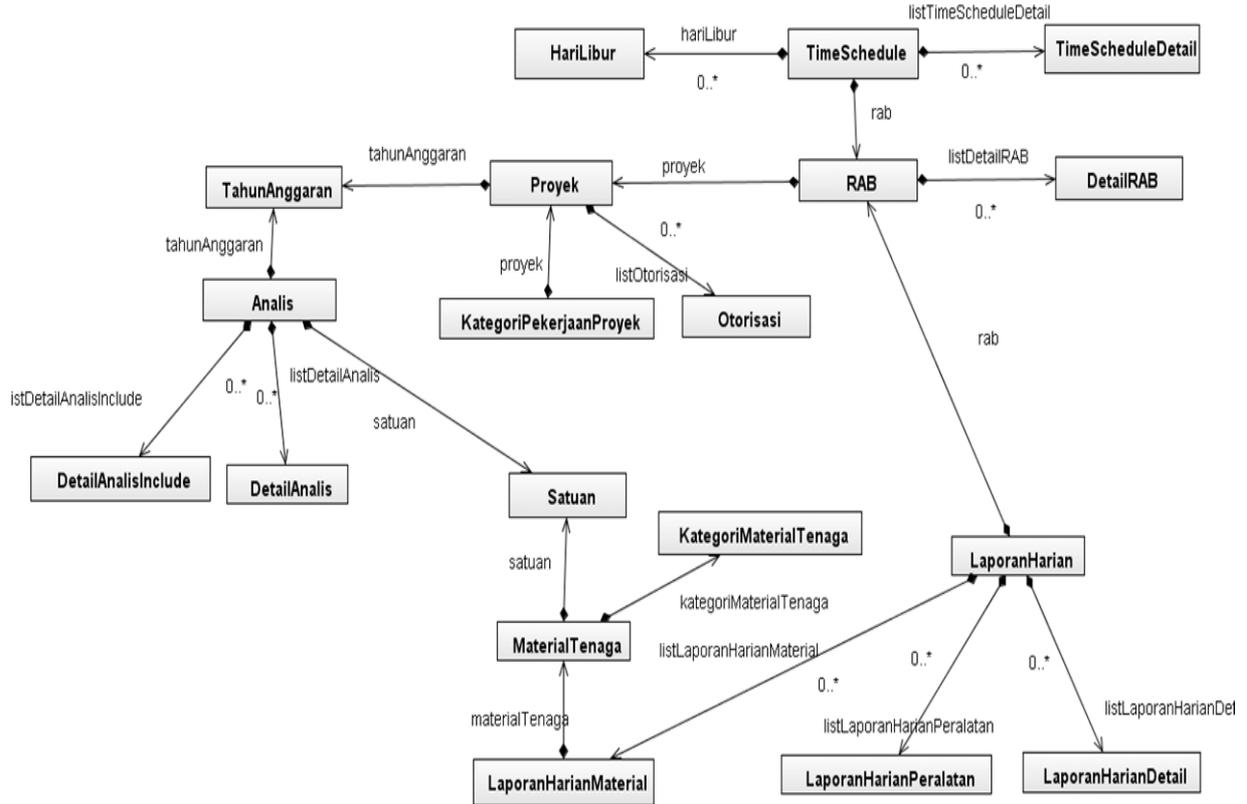
Manajemen-manajemen pengguna, tahun anggaran, kategori material tenaga, manajemen satuan, material tenaga kerja, proyek, *time schedule*, dan sebagainya, memiliki beberapa deskripsi prosesnya masing-masing. Deskripsi-deskripsi proses ini dapat dilihat pada Gambar 6 dan Gambar 7.



Gambar 6. Activity Diagram Admin



Gambar 7. Activity Diagram Non Admin



Gambar 8. Class Diagram AruNG

Class diagram menggambarkan struktur dan deskripsi class, package dan object beserta hubungan yang terjadi antara class yang satu dengan class yang lainnya. Pada artikel ini, aplikasi dirancang menggunakan Hibernate. Dengan Hibernate, class atau objek yang telah dibuat dapat diubah menjadi tabel dalam basis data secara langsung dengan menuliskan file hibernate.cfg.xml-nya. Oleh karena itu, pada tahap perancangan class diagram ini sekaligus menjadi tahap perancangan basis data. Gambar class diagram aplikasi yang dibangun serta hubungan yang terjadi antar class dapat dilihat pada Gambar 8.

adanya beberapa fitur tambahan, yaitu fitur untuk mengatur hari libur proyek, time schedule, laporan harian serta perbandingan time schedule dengan laporan harian. Prototype 1 ini sudah dapat melakukan perhitungan yang akurat pada RAB dan penjabarannya menjadi daftar kebutuhan material dan tenaga, belum terdapat perhitungan nilai bobot serta belum dilakukan pembulatan terhadap total RAB. Selain itu, kekurangan lain yang masih menjadi kendala pada prototype 1 adalah proteksi input yang masih belum diimplementasikan.

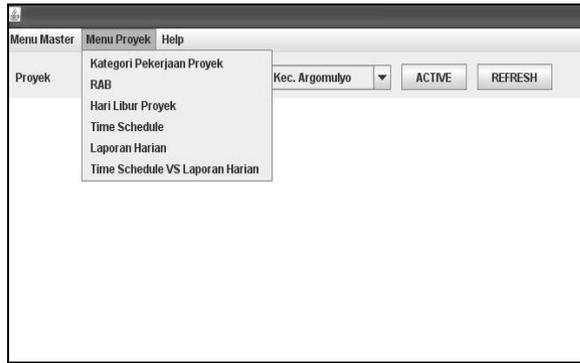
PROTOTYPE YANG DIBANGUN

Sebelum membahas hasil jadi aplikasi yang dibangun, terlebih dahulu dibahas masing-masing prototype yang ada. Prototype 1 direpresentasikan oleh Gambar 9, sedangkan prototype 2 direpresentasikan oleh Gambar 10.

Gambar 9 adalah form home pada RAB prototype 1, pada RAB prototype 1 ini aplikasi hanya memiliki fitur sampai dengan fitur RAB. Fitur-fitur dari prototype 1 ini yaitu data master, RAB sampai dengan membuat daftar kebutuhan material dan tenaga. Setelah dievaluasi, pengguna menginginkan



Gambar 9. Form Home Pada RAB Prototype 1

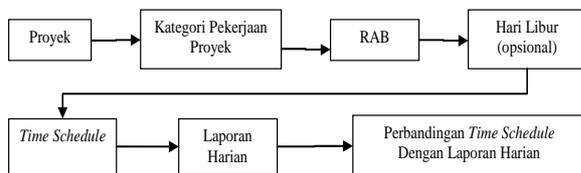


Gambar 10. Form Home Pada RAB Prototype 2

Gambar 10 adalah form home pada RAB prototype 2, pada RAB prototype 2 ini aplikasi sudah memiliki fitur-fitur seperti hari libur proyek, time schedule, laporan harian, serta perbandingan time schedule dengan laporan harian. Pengembangan RAB prototype 2 ini sesuai dengan evaluasi RAB prototype 1. Namun, kekurangan yang masih terdapat pada RAB prototype 2 ini yaitu tampilan yang belum terlalu diperhatikan.

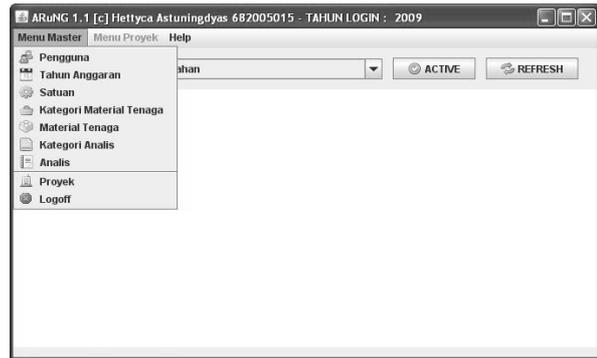
HASIL IMPLEMENTASI APLIKASI RAB

Setelah membahas masing-masing prototype yang telah dikembangkan, penulis memberikan pembahasan tentang pembuatan RAB menggunakan hasil jadi dari aplikasi yang telah dibangun. Alur pembuatan RAB dapat dilihat pada Gambar 11.



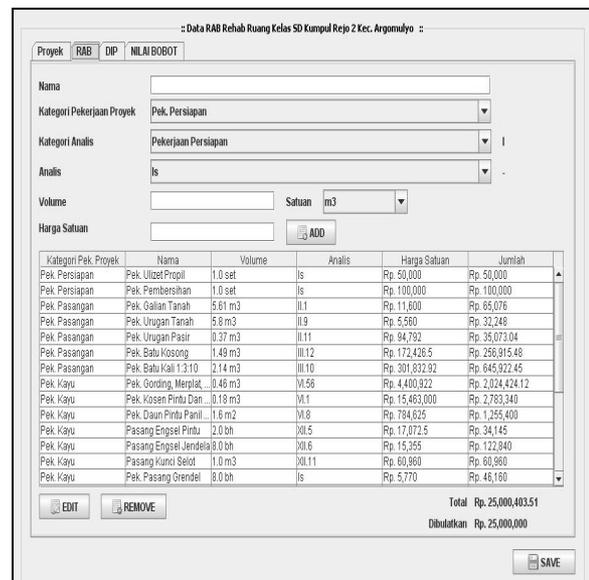
Gambar 11. Alur Pembuatan RAB dengan ARuNG

Adapun alur pembuatan RAB seperti yang tertera pada Gambar 11, diawali dengan membuat data proyek terlebih dahulu kemudian dari proyek tersebut dibuat data tentang kategori-kategori pekerjaan proyek. Setelah data kategori proyek dibuat, langkah selanjutnya ialah membuat data tentang RAB yang berisi macam-macam pekerjaan yang ada, beserta volume, serta analisis yang digunakan. Ketika RAB dibuat secara otomatis, aplikasi akan menghitung daftar kebutuhan material tenaga serta nilai bobot. Nilai bobot merupakan bobot pekerjaan yang berorientasi terhadap harga masing-masing jenis pekerjaan.



Gambar 12. Form Home

Gambar 12 memperlihatkan form utama yang merupakan pusat aplikasi RAB. Gambar 13, Gambar 14 dan Gambar 15 menunjukkan beberapa form dari aplikasi RAB yang telah diimplementasikan. Pada Gambar 13, diperlihatkan form RAB tab RAB di mana pengguna dapat membuat RAB.



Gambar 13. Form RAB Tab RAB

Pada Gambar 13, diperlihatkan form RAB tab DIP yang berisi daftar kebutuhan material serta tenaga sesuai dengan RAB yang telah dibuat sebelumnya. Isi data dari tab DIP ini terbentuk secara otomatis sesuai dengan isi data pada tab RAB.

Pada Gambar 14, diperlihatkan form RAB tab nilai bobot yang berisi nilai bobot suatu pekerjaan. Nilai bobot ini berorientasi terhadap harga masing-masing jenis pekerjaan. Isi data dari tab nilai bobot ini terbentuk secara otomatis sesuai dengan isi data pada tab RAB.

Gambar 14. Form RAB Tab DIP

Gambar 15. Form RAB Tab Nilai Bobot

PENGUJIAN APLIKASI RAB

Pengujian pertama yang dilakukan adalah pengujian keakuratan RAB dari aplikasi dengan RAB yang dibuat dengan perhitungan manual. Pengujian

kedua adalah keakuratan penjabaran RAB menjadi daftar kebutuhan material dan tenaga dibandingkan dengan penjabaran RAB yang dibuat dengan perhitungan manual. Pada pengujian pertama, hal yang dibandingkan adalah keakuratan jumlah biaya dari komponen-komponen yang membentuk proyek, karena hal ini sangat krusial di dalam pembuatan RAB, sehingga kesalahan komponen biaya sekecil apapun dapat mempengaruhi hasil dari RAB secara keseluruhan. Pada pengujian kedua yaitu penjabaran RAB menjadi daftar kebutuhan material dan tenaga.

Besarnya total RAB beserta nama proyek serta selisihnya antara daftar kebutuhan material dan tenaga dengan perhitungan manual dapat dilihat pada Tabel 1. Hasil yang ditampilkan pada tabel merupakan total RAB serta total daftar kebutuhan material dan tenaga yang sudah dilakukan pembulatan yaitu pembulatan secara ribuan.

Dari hasil pengujian yang telah dilakukan dapat disimpulkan bahwa aplikasi yang dibangun sudah menghasilkan keluaran yang akurat, selisih yang terjadi adalah perbedaan minor yang terjadi karena adanya pembulatan.

KESIMPULAN

Berdasarkan pembahasan yang telah dilakukan pada bagian-bagian sebelumnya, maka dapat ditarik kesimpulan yaitu perancangan aplikasi yang mampu menjabarkan RAB untuk pembangunan dan pemeliharaan/rehabilitasi gedung menjadi daftar kebutuhan material dan tenaga dengan teliti, cepat, akurat, dan cermat dapat diwujudkan dengan perangkat lunak yang dibuat menggunakan bahasa pemrograman Java dan menggunakan *Hibernate* sebagai *tool* ORM (*Object Relational Mapping*), serta MySQL sebagai server basis datanya.

Tabel 1. Perhitungan total RAB dan daftar kebutuhan material serta tenaga secara manual dibandingkan menggunakan AruNG

Nama Proyek	Total RAB (Rp)	Total Daftar Kebutuhan Material Dan Tenaga Secara Manual (Rp)	Total Daftar Kebutuhan Material Dan Tenaga Menggunakan Perangkat Lunak Arung (Rp)	Selisih (Rp)	Selisih (%)
Rehabilitasi Kantor Kecamatan Kota Salatiga 2009	79.097.000,00	79.039.000,00	79.097.000,00	58.000,00	0,073%
Rehabilitasi Kantor Kelurahan	53.806.000,00	53.806.000,00	53.806.000,00	0,00	0%
Rehabilitasi Ruang Rawat Inap Rumah Sakit	25.071.000,00	25.017.000,00	25.071.000,00	54.000,00	0.215%
Rehabilitasi Gedung Perawatan Jalan Terpadu Rumah Sakit	4.912.000,00	4.912.000,00	4.912.000,00	0,00	0%

DAFTAR PUSTAKA

1. Bagian Administrasi Keuangan, Sekretariat Daerah Kota Salatiga, 2008, *Peraturan Walikota Salatiga Nomor 35 Tahun 2008 tentang Standarisasi Indeks Biaya Di Lingkungan Pemerintah Kota Salatiga*. Salatiga.
2. Ibrahim, B., 1993, *Rencana dan Estimate Real of Cost*. Bumi Aksara: Jakarta.
3. Sastraatmadja, A. S., 1984, *Analisa Anggaran Biaya dan Pelaksana*. Nova: Bandung.
4. Mukomoko, J. A. 1987, *Dasar Penyusunan Anggaran Biaya Bangunan Metode BOW*. Gaya Media Pratama, Jakarta.
5. Thamura, F., Haryanto, L., dan Muhardin, E., 2006. *Cara Cepat Mengembangkan Solusi Java Enterprise dengan Arsitektur MVC (Struts2, Spring, Hibernate)*. Jakarta: Penerbit Bambumas.
6. Pressman, R. S., 2007, *Rekayasa Perangkat Lunak*, Yogyakarta: Penerbit Andi.
7. Nugroho, A., 2004. *Analisis dan Perancangan Sistem Informasi Menggunakan Metodologi Berorientasi Objek*. Bandung: Penerbit Informatika.