

PENGEMBANGAN APLIKASI PROJECT TIME MANAGEMENT BERBASIS CLOUD COMPUTING

Arwin Halim

STMIK Mikroskil

Jl. Thamrin No. 122, 124, 140 Medan 20212

arwin@mikroskil.ac.id

Abstrak

Perkembangan teknologi informasi telah meningkatkan kebutuhan terhadap informasi. Informasi yang *up-to-date* dengan dukungan teknologi internet telah menjadi bagian penting dalam menjalankan aktivitas. Tren aplikasi yang sedang berkembang adalah aplikasi yang diakses secara berkelompok atau sosial. Aplikasi *project time management* merupakan salah satu contoh aplikasi yang digunakan secara berkelompok, dan menjadi media monitoring dan kontrol proyek. Penentuan jadwal, progress pelaksanaan, dan penyelesaian masalah yang terjadi, memiliki tujuan untuk menyelesaikan proyek tepat waktu. Aplikasi *project time management* memiliki ketergantungan terhadap waktu (*realtime*), sehingga memerlukan dukungan sistem yang tinggi dan dalam implementasinya memerlukan biaya yang mahal. Salah satu solusinya adalah membangun aplikasi dengan model *cloud computing* yang telah memanfaatkan kemampuan infrastruktur penyedia layanan, sehingga pengembang dapat menghasilkan layanan yang bersifat *scalable* dan *fault tolerant*. Aplikasi penelitian dirancang menggunakan UML diagram dan menghasilkan aplikasi web *project time management* dengan *cloud platform* dari Google. Pada penjadwalan, *ganttt-chart* digunakan sebagai alat untuk merepresentasikan jadwal proyek dan *actual progress* untuk setiap aktivitas tim, serta *PERT* digunakan untuk memperkirakan waktu penyelesaian proyek dalam usaha pengontrolan dari segi waktu, sehingga manajer proyek dapat dengan cepat memberi instruksi ketika terjadi permasalahan.

Kata kunci : *cloud computing, gantt chart, google platform, PERT, project time management*

1. Pendahuluan

Perkembangan perangkat lunak memberikan perubahan-perubahan besar dalam arsitektur komputer, penambahan data pada memori dan kapasitas penyimpanan, serta variasi pilihan masukan dan keluaran yang luas. Semua perubahan telah mempercepat sistem berbasis komputer yang sudah kompleks dan canggih. Kecanggihan dan kompleksitas tersebut dapat memberi hasil yang mengagumkan jika sistem berjalan, tetapi menjadi masalah besar bagi mereka yang harus membangun dan memelihara sistem yang kompleks. Selain itu, faktor lain seperti *availabilty*, *mobility* dan *interoperability* telah menjadi tren kebutuhan pengguna [1]. Faktor-faktor tersebut muncul karena aplikasi telah berkembang dalam skala besar, baik dari segi jumlah pengguna maupun jangkauan yang tersebar. Salah satu contoh aplikasinya adalah aplikasi *project time management*. Aplikasi tersebut digunakan untuk membantu menjawab masalah manajemen waktu. Masalah *project time management* adalah keterlambatan waktu penyelesaian proyek [2]. Salah satu penyebabnya adalah kegagalan manajer proyek mengetahui bahwa proyek telah ketinggalan dari jadwal yang ada dan kurangnya tindakan untuk memecahkan masalah tersebut. Aplikasi *project time management* digunakan pada berbagai bidang dan mungkin memiliki pengguna dalam bentuk tim yang tersebar. Aplikasi tersebut dapat dikembangkan dengan berbagai model. Namun, seiring dengan perkembangan teknologi perangkat keras (tidak hanya berupa *personal computer*) dan menurunnya biaya

konektivitas, *cloud computing* menjadi alternatif yang menarik [1]. Aplikasi *cloud computing* diletakkan pada infrastruktur *grid* yang memberikan kelebihan seperti *scalability*, *fault tolerant*, dan menggunakan sumber daya bersama [3]. Aplikasi web dengan model *cloud computing* memberi kemudahan akses dan tidak memerlukan proses instalasi, serta mengurangi pemeliharaan terhadap aplikasi karena proses *update* dan *upgrade* dilakukan secara terpusat.

Tujuan dari penelitian adalah mengembangkan aplikasi web *project time management* sebagai media monitoring dan kontrol proyek sehingga proyek dapat diselesaikan tepat waktu. Aplikasi tersebut dikembangkan dengan model *cloud computing*, sehingga aplikasi dapat memperoleh semua keunggulan dari model tersebut. Manfaat yang diperoleh dari penelitian adalah tersedianya media untuk komunikasi antara manajer proyek dan anggota tim, memberikan arahan kerja bagi tim yang terpisah secara geografis, dan melihat kinerja dari masing-masing anggota tim dalam penyelesaian proyek. Penelitian ini juga memberikan gambaran bagi pengembang sistem dalam merancang aplikasi berbasis *cloud computing*.

Aplikasi web berbasis *cloud computing* dikembangkan dengan menggunakan *cloud platform* yang telah disediakan oleh penyedia layanan. Penelitian ini menggunakan *cloud platform* yang disediakan oleh Google, yaitu Java SDK Google App Engine, Datastore App Engine, dan Google Web Toolkit Framework. Aplikasi mampu menghasilkan *Gantt Chart* untuk menampilkan jadwal pelaksanaan proyek dan menggunakan teknik *Program Evaluation and Review Technique* (PERT) untuk menentukan perkiraan durasi penyelesaian proyek. Aplikasi ini belum mendukung terjadinya pemecahan terhadap satu aktivitas dan kolaborasi antar proyek.

2. Kajian Pustaka

2.1 Internet

Internet merupakan sebuah kumpulan yang terdiri dari ribuan jaringan komputer dan jutaan komputer pribadi yang dikelola secara bebas. Internet memungkinkan komunikasi antarkomputer dengan menggunakan *Transmission Control Protocol* atau *Internet Protocol* (TCP/IP) yang didukung media komunikasi, seperti satelit dan radio paket sehingga jarak menjadi tidak terbatas [4]. Layanan yang banyak dimanfaatkan dalam internet adalah *World wide web* atau web. Perkembangan layanan *World wide web* semakin pesat setelah terjadinya *dot-com bubble* pada tahun 2001. Konsep web 2.0 dimulai dalam sesi *brainstorming* antara tim O'Reilly dan MediaLive International. Web 2.0 merupakan revolusi bisnis dalam industri komputer yang disebabkan oleh pergeseran internet sebagai *platform* dan adanya usaha untuk memahami aturan-aturan untuk keberhasilan pada *platform* baru [5].

2.2 Cloud Computing

Perkembangan web yang pesat, kemampuan perangkat keras yang meningkat, biaya konektivitas yang menurun, dan meningkatnya kebutuhan untuk menjaga ketersediaan serta meningkatkan layanan untuk banyak pengguna aplikasi web menjadi salah satu pemicu pemanfaatan konsep *cloud computing*. *Cloud computing* merupakan realisasi dari internet (*cloud*) berdasarkan pengembangan dan penggunaan teknologi komputer (*compute*) yang disediakan oleh suatu ekosistem penyedia layanan. *Cloud* memungkinkan pengguna dalam suatu organisasi untuk menjalankan dan *men-deploy* aplikasi dan *virtual datacenter* dengan sumber daya yang mungkin berada pada beberapa lokasi di dalam atau di luar dari organisasi [1]. Biaya yang diperlukan untuk mendapatkan layanan dihitung berdasarkan sumber daya yang digunakan seperti dukungan komputasi dari perangkat keras, dukungan terhadap

datacenter dan lain-lain. Karakteristik dari *cloud computing* adalah *incremental scalability*, *agility*, keandalan dan *fault tolerant*, berorientasi pada layanan, berorientasi pada *utility*, dan kebersamaan [3].

Komponen dari *cloud computing* [3] terdiri dari:

1. *Infrastructure*: infrastruktur yang menunjukkan konsep '*hardware as a service*' yaitu perangkat keras yang digunakan secara bersama sebagai layanan.
2. *Storage*: konsep yang menunjukkan pemisahan data dari pengolahan dan penyimpanan yang terletak pada tempat yang berjauhan. Layanan ini juga disebut layanan *database*.
3. *Platform*: layanan untuk *men-deploy* aplikasi dan mengelola perangkat keras dan perangkat lunak yang diperlukan.
4. *Application*: konsep perangkat lunak yang menawarkan arsitektur yang menghilangkan kebutuhan untuk menginstal, menjalankan dan memelihara aplikasi pada pengguna *desktop*.
5. *Service*: bagian yang *independen* dari perangkat lunak yang dapat digunakan bersama-sama untuk mendukung interaksi antar aplikasi melalui jaringan.
6. *Client*: perangkat keras dan lunak yang memanfaatkan layanan *cloud computing* melalui jaringan. Perangkat *client* dapat berupa web *browser*, laptop, PC, dan lain-lain.

2.3 Google Platform

2.3.1 Google App Engine

Google App Engine [6] merupakan salah satu *cloud platform* yang memungkinkan pengembangan suatu aplikasi web yang berjalan dengan menggunakan infrastruktur Google. Aplikasi dari App Engine mudah dibangun, mudah dipelihara, dan bersifat *scalable* sesuai kebutuhan lalu lintas dan media penyimpanan. App Engine tidak memerlukan *server* untuk dipelihara, pengembang hanya perlu meng-*upload* aplikasi yang telah dibuat dengan SDK App Engine pada Google, dan aplikasi dapat memanfaatkan infrastruktur Google. Google App Engine akan mempermudah pengembang membangun aplikasi yang terpercaya bahkan pada lingkungan yang memiliki beban yang berat dan jumlah data yang banyak. Fitur-fitur dari App Engine meliputi:

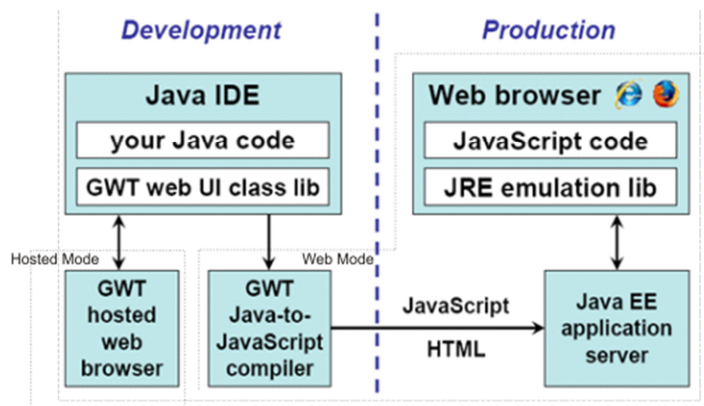
1. Layanan web dinamis dengan dukungan penuh terhadap teknologi web.
2. Media penyimpanan yang tetap dengan dukungan *query*, pengurutan, dan transaksi.
3. *Scale* secara otomatis dan *load balancing*.
4. API untuk otentikasi pengguna dan mengirim email dengan Google Account.
5. Fitur lengkap yang dapat mensimulasikan *Google App Engine* pada komputer lokal.
6. Tugas-tugas yang terjadwal yang dipicu pada waktu tertentu atau pada interval teratur.

2.3.2 Datastore App Engine

App Engine [6] memiliki suatu layanan untuk menyimpan data secara terdistribusi dan menyediakan fasilitas *query* dan transaksi pada aplikasi. Pendistribusian server *web* dan *datastore* yang terdistribusi dapat mendukung pertumbuhan lalu lintas *web* dan *datastore* yang terdistribusi dapat mendukung ketersediaan data pada aplikasi. Datastore App Engine tidak seperti *database relational* yang tradisional. Datastore App Engine menggunakan *data object*. *Data object* menggabungkan kumpulan sifat dari entitas. Proses *query* dilakukan untuk mengambil entitas-entitas berdasarkan beberapa filter dan melakukan pengurutan terhadap nilai berdasarkan sifat-sifatnya. *Datastore* entitas bersifat *schemaless* berarti tersedia fitur untuk mendefinisikan struktur *datastore* yang dibuat dalam aplikasi. Selain itu, *datastore App Engine* juga mendukung nilai yang tetap pada tipe data dan properti entitas.

2.3.3 Google Web Toolkit

Google web toolkit (GWT) [7] merupakan *tool* yang dapat dipakai pengembang web untuk membuat dan memelihara aplikasi AJAX yang kompleks dengan bahasa pemrograman java. GWT memiliki lisensi Apache versi 2.0 dan bersifat *open source*. Pengembang web dengan GWT dapat menerapkan prinsip pemrograman berbasis objek dan menggunakan editor java untuk membuat aplikasi web. Kompilator GWT dapat menerjemahkan kode Java menjadi berkas javascript yang dikenal oleh browser web. Komponen umum dari GWT adalah *GWT Java-to-JavaScript Compiler*, *GWT Hosted Web Browser*, *JRE emulation library*, dan *GWT web user interface class library*. Gambaran umum aplikasi GWT dapat dilihat pada Gambar 1.



Gambar 1. Gambaran Cara Kerja Aplikasi GWT

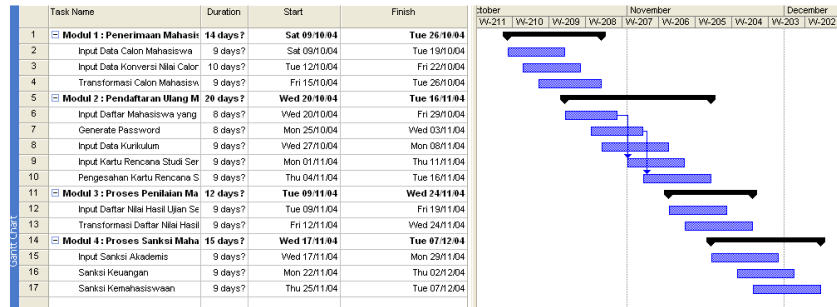
2.4 Manajemen Proyek

Manajemen proyek terdiri dari kata manajemen dan proyek. Manajemen adalah pengaturan aktivitas-aktivitas kerja sehingga dapat selesai secara efisien dan efektif dengan dan melalui orang lain. Efisiensi digambarkan sebagai “melakukan pekerjaan dengan benar” artinya tidak memboroskan sumber daya. Efektivitas digambarkan sebagai melakukan pekerjaan yang benar, yaitu aktivitas-aktivitas kerja yang membantu organisasi mencapai sasaran. Parameter penting yang menjadi *triple constraint* dalam penyelenggaraan proyek yang sering diasosiasikan sebagai sasaran dari proyek dan dapat menjadi indikator keberhasilan dari proyek adalah biaya proyek, mutu pekerjaan, dan waktu penyelesaian [8].

2.5 Project Time Management

Project Time Management merupakan salah satu bidang pengetahuan manajemen proyek yang berkaitan dengan waktu penyelesaian proyek. Manajer menyebutkan, proyek yang diselesaikan tepat waktu sebagai suatu tantangan yang besar. Manajer menyebutkan persoalan pembuatan jadwal disebabkan terjadinya konflik jadwal sepanjang pengerjaan proyek. Mungkin alasan terjadinya masalah dalam penjadwalan adalah hal umum yaitu waktu mudah diukur. Secara sederhana, *project time management* melibatkan proses-proses yang diperlukan dalam memastikan penyelesaian proyek yang tepat waktu. Proses-proses yang diperlukan dalam *project time management* adalah *activity definition*, *activity sequencing*, *activity resource*, *activity duration estimating*, *schedule development*, dan *schedule control*. [2]

Jadwal pengembangan proyek dapat digambarkan dengan *Gantt chart*. *Gantt chart* merupakan ringkasan sistem yang menunjukkan urutan dan waktu dari aktivitas pada proses. *Gantt chart* digambarkan dengan diagram batang pekerjaan dan *milestone*. Gambar 2 menunjukkan jadwal pengembangan proyek dengan *gant chart*.



Gambar 2. Jadwal Pengembangan Proyek dengan Gantt Chart

Penentuan jadwal pengembangan proyek diperlukan untuk menentukan awal dan akhir dari proyek. Teknik *Program Evaluation and Review Technique* (PERT) merupakan salah satu teknik untuk memperkirakan durasi penyelesaian proyek. Prosedur penjadwalan PERT bersifat probabilistic [9] yaitu durasi setiap kegiatan ditentukan oleh tiga perkiraan durasi kegiatan, yaitu *optimistic estimate* (t_o), *pessimistic estimate* (t_p), dan *most likely estimate* (t_m). Dengan menggunakan tiga perkiraan durasi waktu, perkiraan pengerjaan aktivitas menjadi lebih realistis. Durasi efektif dari setiap kegiatan (t_e) merupakan durasi yang didapat dari perhitungan tiga perkiraan durasi untuk masing-masing kegiatan. Perhitungan durasi efektif ditunjukkan oleh Persamaan 1.

$$t_e = (t_o + 4t_m + t_p) / 6 \quad (1)$$

Nilai standar deviasi dari setiap kegiatan digunakan untuk menghitung varian dari kegiatan. Perhitungan nilai varian ditunjukkan oleh Persamaan 2.

$$D = (t_p - t_o) / 6 \quad (2)$$

3. Metode Penelitian

3.1 Analisis

Analisis aplikasi *project time management* berbasis *cloud computing* dapat dilakukan dengan cara menentukan persyaratan sistem antara lain: persyaratan fungsional dan persyaratan nonfungsional. Persyaratan fungsional sistem direpresentasikan dengan *Unified Modeling Language (UML) use case diagram*. *Use case diagram* pada aplikasi *project time management* terdiri dari tiga aktor yaitu calon anggota, anggota tim, dan manajer proyek. Tabel 1 menjelaskan deskripsi dari aktor pada *use case diagram*. *Use case diagram* sistem dapat dilihat pada Gambar 3.

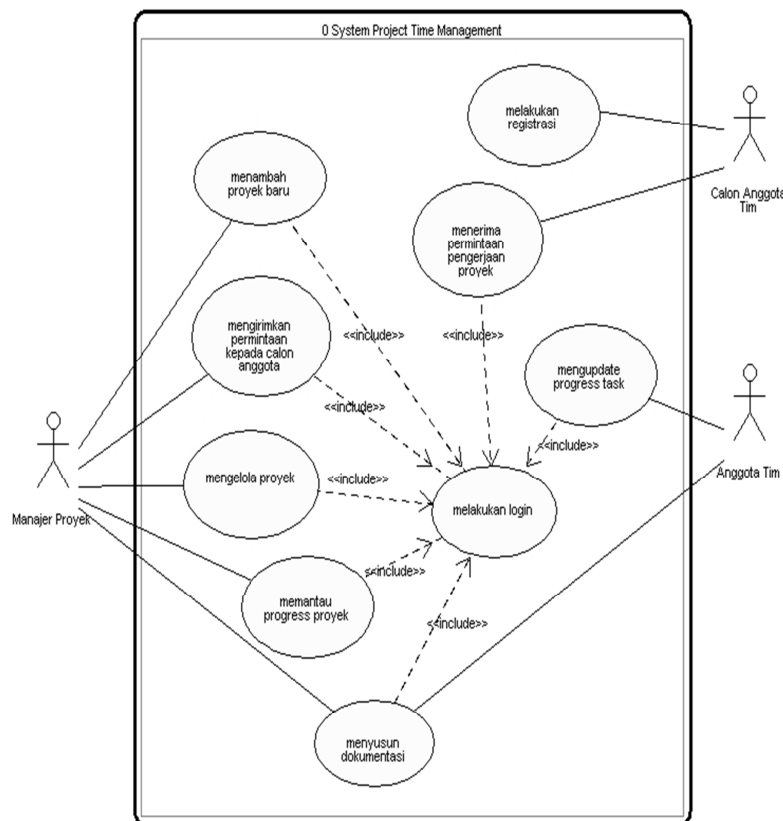
Persyaratan nonfungsional sistem direpresentasikan dengan kerangka PIECES sebagai berikut:

1. *Performance*. Sistem harus dapat memproses setiap aktivitas yang dimasukkan pengguna secara real-time.
2. *Information*. Manajer proyek dan anggota tim dapat memperoleh data paling *up-to-date* mengenai *progress* proyek secara keseluruhan secara akurat.
3. *Economy*. Sistem memanfaatkan *cloud computing* sehingga bersifat *pay-per-use* atau *free*, sehingga mengurangi kendala biaya dalam penggunaan aplikasi dan mengubah biaya modal menjadi biaya operasional.
4. *Control*. Sistem menyediakan role untuk pengguna sebagai manajer atau anggota. Manajer proyek mampu melakukan kontrol dengan memantau *progress* aktivitas yang telah diselesaikan tim dan memberi instruksi pada anggota tim.

5. *Efficiency*. Manajer proyek dan anggota tim tidak perlu melakukan proses instalasi apapun untuk menggunakan aplikasi, karena aplikasi berbasis web.
6. *Service*. Sistem disediakan dalam bentuk aplikasi web sehingga memberi kemudahan tim mengakses dan meng-*update progress* penyelesaian aktivitas proyek.

Tabel 1. Deskripsi Aktor Aplikasi Project Time Management

Aktor	Deskripsi
Calon anggota	Orang yang berpotensi untuk mengerjakan proyek. Calon anggota harus memiliki <i>account gmail</i> untuk dapat menggunakan aplikasi.
Anggota tim	Orang yang telah menerima undangan pengerjaan proyek dan memiliki tanggung jawab untuk menyelesaikan aktivitas yang telah ditentukan.
Manajer proyek	Orang yang membuat proyek baru, mendefinisikan semua atribut proyek, dan memiliki peran untuk mengoordinasikan serta memantau pelaksanaan proyek.

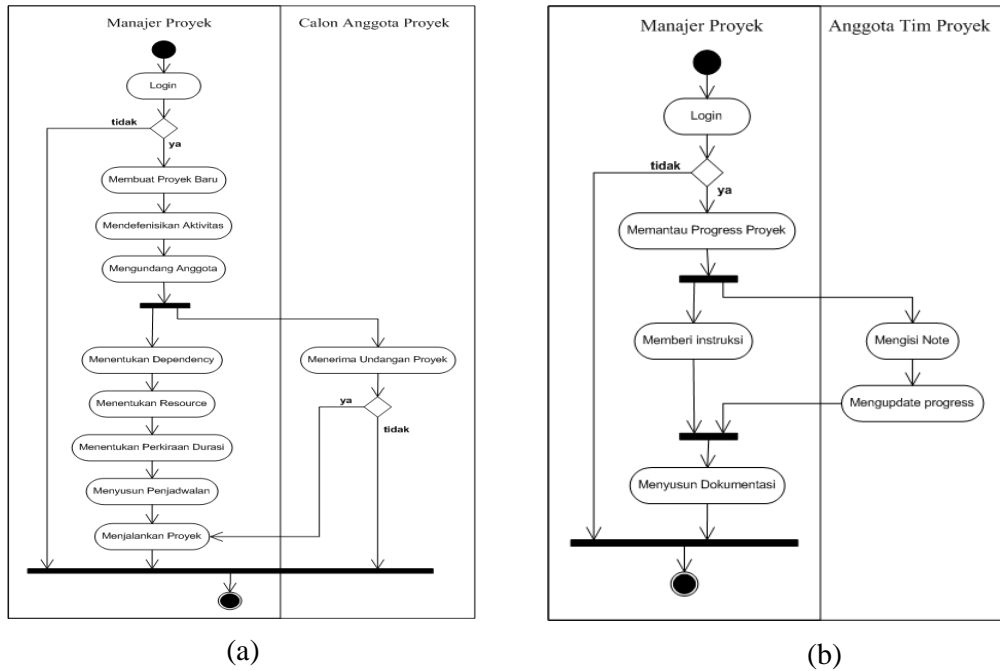


Gambar 3. UML Use Case Diagram untuk Aplikasi Project Time Management

3.2 Perancangan

3.2.1 Perancangan Proses

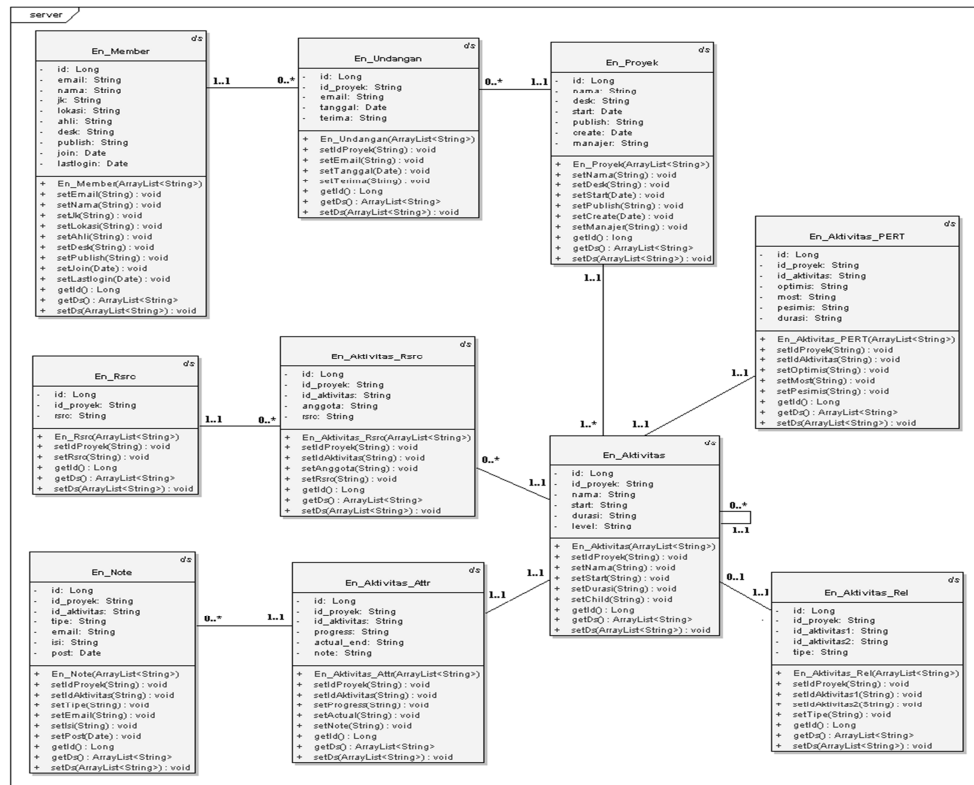
Secara garis besar, sistem *project time management* terdiri dari dua bagian utama yaitu menyusun jadwal proyek dan melakukan proses memantau pelaksanaan proyek. Alur kerja dan logika proses pada sistem dapat digambarkan dengan UML activity diagram. Gambar 4(a) menunjukkan *activity diagram* untuk proses penyusunan jadwal proyek dan Gambar 4(b) menunjukkan *activity diagram* untuk proses memantau pelaksanaan proyek.



Gambar 4. Activity Diagram Aplikasi Project Time Management

3.2.2 Perancangan Data

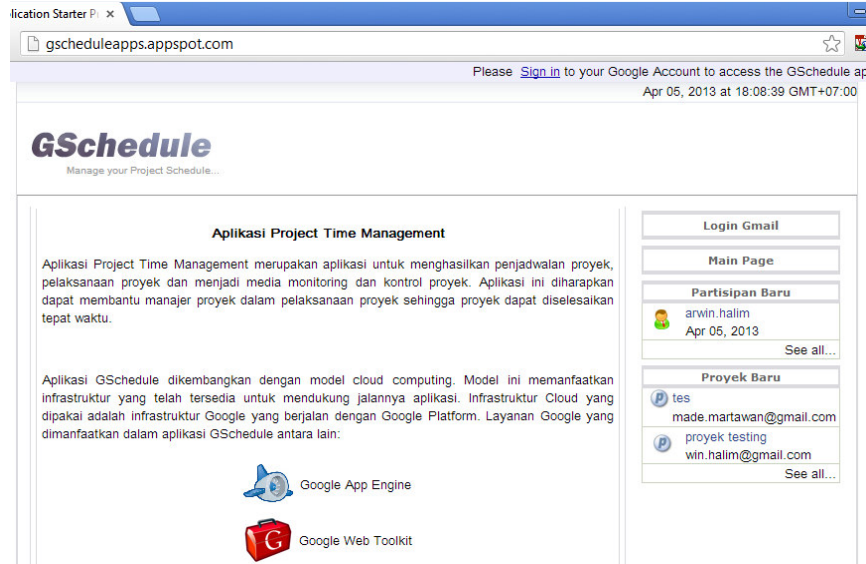
Data pada sistem dapat digambarkan dengan menggunakan UML class diagram. Gambar 5 menunjukkan class diagram untuk aplikasi project time management.



Gambar 5. UML Class Diagram Aplikasi Project Time Management

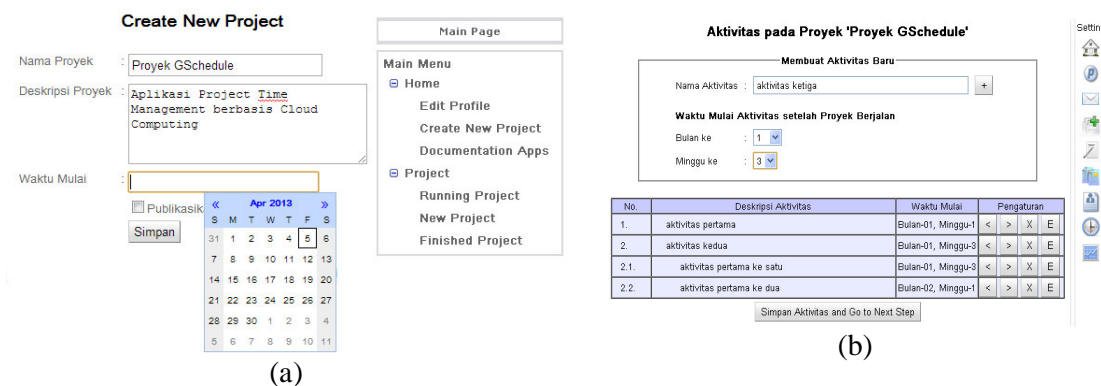
4. Hasil dan Pembahasan

Setelah penulis melakukan penelitian dan perancangan sistem, hasil yang dikembangkan berupa sebuah aplikasi web yang menggunakan *Google App Engine*, *Datastore App Engine* dan *Google Web Toolkit*, yang diberi nama Aplikasi *GSchedule*. Aplikasi *GSchedule* dapat diakses pada <http://gscheduleapps.appspot.com>. Gambar 6 menunjukkan tampilan awal hasil dari aplikasi project time management berbasis *cloud computing*.



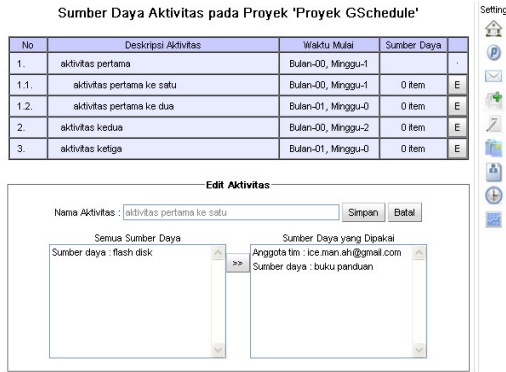
Gambar 6. Tampilan Awal Aplikasi GSchedule

Halaman awal website terdiri dari bagian pengenalan website dan menu-menu pilihan seperti menu login ke Google, menu daftar member atau proyek baru. Menu-menu pada halaman awal dapat diakses secara bebas dan tidak memerlukan proses autentikasi pengguna. Pengguna dapat login menggunakan account Google, karena aplikasi ini sudah terintegrasi dengan layanan Account pada Google. Setelah melakukan login, pengguna dapat memulai untuk membuat proyek baru seperti yang terlihat pada Gambar 7(a), kemudian melakukan proses manajemen waktu dengan membuat jadwal pengerjaan proyek yang terlihat pada Gambar 7(b)-(f). Aplikasi menyediakan estimated dan actual progress dalam bentuk angka dan Gantt-chart, serta note dan instruksi sebagai bagian dari proses kontrol dan pemantauan pelaksanaan proyek, terlihat pada Gambar 7(g). Setelah proyek selesai dikerjakan, aplikasi *GSchedule* menyediakan dokumentasi akhir pengerjaan proyek untuk setiap aktivitas, yang terlihat pada Gambar 7(h).

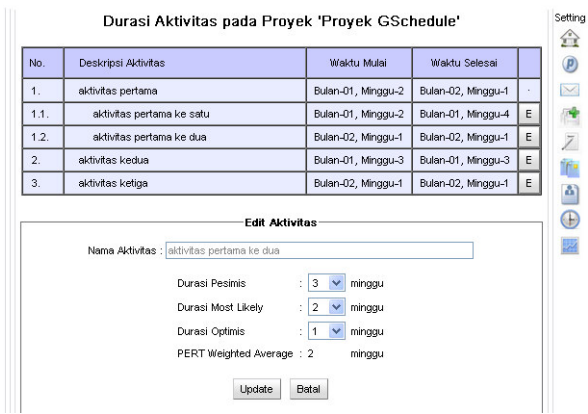




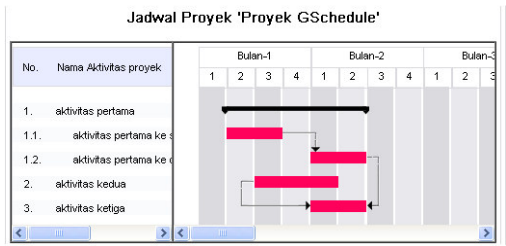
(c)



(d)



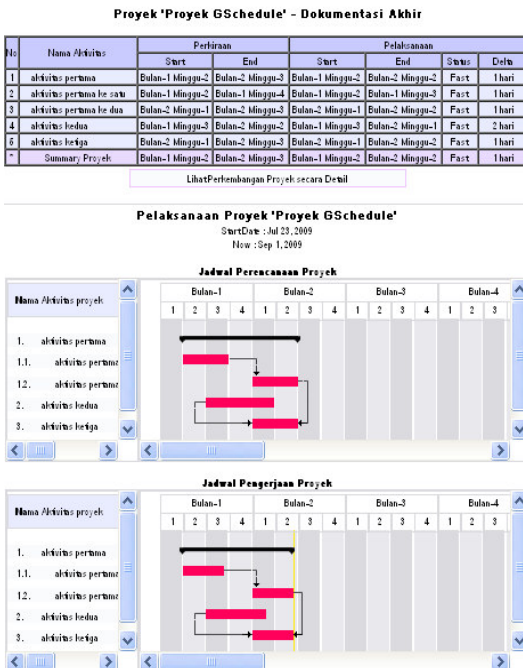
(e)



(f)



(g)



(h)

Gambar 7. Antar Muka Aplikasi GSchedule: (a) Membuat Proyek Baru (b) Definisi Aktivitas (c) Urutan Aktivitas (d) Sumber daya Aktivitas (e) Estimasi Waktu Aktivitas (f) Penjadwalan Proyek (g) Kontrol dan Monitoring Penjadwalan (h) Dokumentasi Akhir Proyek

5. Kesimpulan

Kesimpulan yang dapat ditarik dari hasil implementasi aplikasi GSchedule berbasis *cloud platform* dari Google antara lain:

1. Aplikasi GSchedule mendukung perencanaan jadwal proyek dengan mendefinisikan atribut dan proses kontrol terhadap jadwal pelaksanaan proyek, sehingga dapat memberikan arah kerja bagi seluruh tim yang tersebar secara geografis.
2. Aplikasi GSchedule yang dibangun dengan *cloud platform* mampu menggunakan infrastruktur Google, sehingga semua layanan Google seperti *Google Account*, *send mail services*, dan *datastore* dapat diintegrasikan pada aplikasi.
3. Setiap *query* dari *datastore App Engine* menggunakan indeks, belum mendukung hubungan banyak ke banyak, belum mendukung *query aggregat*, dan belum mendukung proses *join*.
4. Aplikasi GSchedule mampu berjalan dengan baik walaupun terjadi perubahan (*upgrade*) pada platform Google, yang berarti *cloud platform* dari Google mendukung kompatibilitas kebelakang.

Referensi

- [1] Velte, T., Velte, A., Elsenpeter, R., 2010, *Cloud Computing, A Practical Approach*, McGraw-Hill, Inc. New York.
- [2] Schwalbe, K., 2006, *Information Technology Project Management Fourth Edition*, Canada, Thomson Course Technology.
- [3] Foster, I., Zhao, Y., Raicu, I., Lu S. 2008, *Cloud Computing and Grid Computing 360-Degree Compare*, IEEE conferences on Grid Computing Environments Workshop, pp.1-10.
- [4] Oetomo, B. S. D., Widowo, E., Hartono, E., Prakoso, S., 2007, *Pengantar Teknologi Informasi Internet: Konsep dan Aplikasi*. Andi, Yogyakarta
- [5] O'reilly. 2005. *What is Web2.0 Design Pattern and Business Models for the next Generation of Software*. http://www.ttvanguard.com/ttvanguard_cfmfiles/pdf/dc05/dc05session4003.pdf [12 Feb 2013]
- [6] Anonim, *What is Google App Engine? Google App Engine - Google Code*. <https://developers.google.com/appengine/docs/whatisgoogleappengine?hl=id>. [6 Jan 2013].
- [7] Anonim, *Google Web Toolkit Overview*. <https://developers.google.com/web-toolkit/overview> [12 Feb 2013]
- [8] Kerzner, H., *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*. 2009, ten edition, John Wiley & Sons, New York
- [9] Ervianto, W. I., 2007. *Manajemen proyek konstruksi*. Andi, Yogyakarta.