

Pengembangan Sistem Aplikasi Manajemen Proyek Berbasis Web (Studi Kasus: PT. Swadaya Graha)

Ardian Riftha Dhuha¹, Fajar Pradana², Bayu Priyambadha³

Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹ardian@mitratek.com, ²fajar.p@ub.ac.id, ³bayu_priyambadha@ub.ac.id

Abstrak

PT. Swadaya Graha (Semen Indonesia Group) adalah perusahaan yang bergerak di bidang manufaktur dan jasa konstruksi. Berdasarkan hasil pengamatan dokumen laporan proyek perusahaan dapat disimpulkan bahwa manajer proyek di PT. Swadaya Graha memiliki tugas untuk mengatur jalannya proyek (manajemen proyek). Terdapat permasalahan yang terjadi pada proses manajemen proyek saat ini, diantaranya adalah perlunya alat bantu dalam memudahkan proses perencanaan kegiatan dan biaya proyek, alat untuk mengontrol ketidaksesuaian antara rencana dan realisasi proyek, alat yang dapat mengurangi jumlah penggunaan dokumen laporan yang harus diisi, serta dapat memberikan data-data proyek secara *real-time*. Penelitian ini bertujuan untuk mengembangkan sistem aplikasi manajemen proyek yang dapat membantu perencanaan, melakukan pengawasan proyek, memberikan data laporan secara *real-time*, dan sebagai sarana informasi bagi para pemangku kepentingan. Dibangun dengan menggunakan model pengembangan perangkat lunak bernama *waterfall*, metode pemrograman yang digunakan berbasis *object oriented programming* dan menganut konsep *model view controller*, selain itu terdapat metode utama yang digunakan dalam perencanaan kegiatan proyek yaitu *critical path method*, bertujuan untuk membantu proses penyusunan kegiatan dan penjadwalan proyek. Hasil dari penelitian menunjukkan bahwa pada tahap pengujian terdapat 3 buah pengujian unit yang menghasilkan kesimpulan bahwa fungsi mudah dipahami, diimplementasikan dan di uji, 68 pengujian validasi yang menunjukkan 68 fungsi dapat berjalan dengan baik, dan pengujian kompatibilitas yang menunjukkan sistem dapat digunakan dengan baik pada 8 jenis perambah web.

Kata Kunci: pengembangan, manajemen proyek, *critical path method*, *waterfall*, *object oriented programming*

Abstract

PT. Swadaya Graha (Semen Indonesia Group) is a company engaged in the manufacture and construction services. Based on the observation results of the company project report document can be concluded that the manager project at PT. Swadaya Graha has the task to manage the project (project management). There are problems that occur in the current project management process, including the need for tools to facilitate the process of planning activities and project costs, tools to control the discrepancy between the plan and the realization of the project, tools that can reduce the number of report documents that must be filled, and can provide Project data in *real-time*. This study aims to develop a project management application system that can assist planning, project monitoring, *real-time* reporting data, and as an information tool for stakeholders. Built using a software development model called *waterfall*, the programming method used is *object-oriented programming* and embraces the concept of *view controller model*, besides the main method used in the project activity planning is the *critical path method*, aims to assist the process of project preparation and project scheduling. The results of the research indicate that at the testing stage there are 3 unit tests which resulted in conclusion that the functions are easy to understand, implemented and tested, 68 validation tests showing 68 functions to work properly, and compatibility tests showing the system can be used well on 8 type of web browser.

Keywords: development, project management, *critical path method*, *waterfall*, *object oriented programming*

1. PENDAHULUAN

PT. Swadaya Graha (Semen Indonesia Group) adalah perusahaan yang bergerak di bidang manufaktur dan jasa konstruksi. Berdasarkan hasil pengamatan dokumen laporan proyek PT. Swadaya Graha, dalam mengerjakan setiap proyeknya PT. Swadaya Graha membutuhkan manajer proyek yang bertugas untuk mengatur jalannya proyek. Proses manajemen proyek dibagi dalam 5 tahap yaitu inisiasi, perencanaan, eksekusi, pengawasan, dan penutupan, dimana masing-masing tahap memiliki keterkaitan satu dengan yang lain (Institute, 2008).

Pada tahap inisiasi dan perencanaan, seorang manajer proyek di PT. Swadaya Graha bertanggung jawab dalam melakukan beberapa kegiatan seperti penyusunan kegiatan, merencanakan anggaran biaya, memperkirakan durasi proyek, hingga menentukan sumber daya apa saja yang dibutuhkan. Masalah yang ada pada tahap ini berkaitan dengan waktu dan mutu, manajer proyek tidak memiliki alat bantu khusus dalam menyusun kegiatan beserta rencana anggaran biaya, sehingga umumnya dibuat berdasarkan perkiraan dengan bantuan alat seadanya, proses ini membutuhkan waktu yang relatif lama dan kualitas perencanaannya tergantung dari pengalaman manajer proyek yang menanganinya.

Pada tahap eksekusi, seorang manajer proyek menjalankan setiap aktivitas yang sudah didefinisikan pada tahap inisiasi dan perencanaan, proses implementasi proyek disesuaikan dengan rencana yang dibuat mulai dari susunan kegiatan, anggaran biaya dan sumber daya lain yang dibutuhkan. Permasalahan yang terjadi pada tahap ini berkaitan dengan mutu dan biaya, aktivitas dan penggunaan sumber daya pada proyek dituntut untuk sesuai dengan perencanaan sebelumnya demi menghindari terjadinya ketidaksesuaian. Semakin besar tingkat ketidaksesuaian antara rencana dan realisasi proyek dapat beresiko menimbulkan kerugian bagi perusahaan dan pemangku kepentingan yang terlibat dalam proyek tersebut.

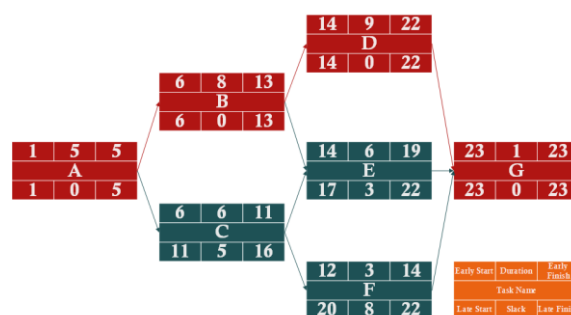
Tahap pengawasan berfungsi untuk meninjau kembali proses yang telah berjalan dan mengevaluasi rencana yang ada sebagai bentuk pengawasan, sistem pelaporannya menggunakan berbagai macam template dokumen elektronik yang telah dibuat sebelumnya oleh perusahaan,

selanjutnya template tersebut diisi oleh pemangku kepentingan terkait dan menyerahkan laporan ke perusahaan secara berkala. Permasalahan yang terjadi pada tahap ini berkaitan dengan waktu, mutu, dan biaya. Data laporan saat ini relatif tidak akurat karena bukan data yang bersifat *real-time*, banyaknya jumlah dokumen laporan yang harus diisi oleh para pemangku kepentingan, serta terdapat resiko terjadinya kesalahan perhitungan biaya dalam penulisan laporan.

Solusi yang dapat digunakan adalah dengan memanfaatkan penggunaan teknologi informasi terintegrasi melalui sistem informasi manajemen proyek berbasis web yang dikombinasikan dengan *Critical Path Method* (CPM) untuk membantu proses perencanaan proyek, dengan adanya sistem aplikasi manajemen proyek ini dapat membantu dalam melakukan perencanaan, meminimalisir terjadinya ketidaksesuaian pada rencana dan realisasi proyek, mengoptimalkan proses perhitungan biaya, serta memudahkan proses pengisian dokumen laporan saat ini dengan cara mengurangi jumlah dokumen yang harus diisi oleh para pemangku kepentingan.

2. METODE

Jalur kritis (*critical path*) adalah sebuah rangkaian aktivitas-aktivitas dari sebuah proyek yang tidak bisa ditunda waktu pelaksanaannya dan menunjukkan hubungan yang saling berkaitan satu dengan yang lain (Heizer dan Render, 2006). CPM adalah teknik dalam menganalisa jaringan kegiatan/aktivitas-aktivitas pada saat menjalankan proyek dalam rangka memprediksi durasi total pengerjaan proyek serta menentukan jalur kritis yang terdapat pada suatu proyek, CPM sendiri berupa algoritma berbasis matematika untuk menghitung penjadwalan sekelompok aktivitas proyek tertentu. Berikut contoh CPM yang ditampilkan pada Gambar 1.



Gambar 1. Contoh CPM
Sumber: Mubarak (2010)

Terdapat 2 tahap perhitungan CPM untuk menentukan jalur kritis (*critical path*), yaitu hitungan maju dan hitungan mundur (Mubarak, 2010):

1. Hitungan maju (*forward pass*), dimulai dari Start (*initial event*) menuju Finish (*terminal event*) untuk menghitung waktu penyelesaian tercepat suatu kegiatan (EF) dan waktu tercepat terjadinya kegiatan (ES).

- a. Waktu tercepat terjadinya kegiatan (ES) didapatkan dengan rumus :

$$ES(j) = \max(EF(i)) \quad (1)$$

- b. Suatu kegiatan baru dapat dimulai bila kegiatan yang mendahuluinya (*predecessor*) telah selesai. Waktu selesai paling awal suatu kegiatan (EF) sama dengan waktu mulai paling awal (ES), ditambah dengan kurun waktu kegiatan yang mendahuluinya (Dur).

$$EF(j) = ES(j) + Dur(j) \quad (2)$$

2. Hitungan mundur (*backward pass*), dimulai dari Finish menuju Start untuk mengidentifikasi saat paling lambat terjadinya suatu kegiatan (LF) dan waktu paling lambat terjadinya suatu kegiatan (LS).

- a. Waktu selesai paling akhir suatu kegiatan (LF) adalah waktu mulai minimum paling akhir dari kegiatan setelahnya.

$$LF(j) = \min(LS(k)) \quad (3)$$

- b. Waktu mulai paling akhir suatu kegiatan (LS) sama dengan waktu selesai paling akhir (LF) dikurangi kurun waktu berlangsungnya kegiatan yang bersangkutan (Dur).

$$LS(j) = LF(j) - Dur(j) \quad (4)$$

Apabila hitungan maju dan hitungan mundur selesai dilakukan maka dapat diperoleh nilai *slack* atau *total float* (TF), *slack* adalah kelonggaran waktu atau elastisitas durasi suatu kegiatan pada proyek, jika nilai *slack* pada suatu kegiatan sama dengan 0 menandakan kegiatan tersebut adalah kegiatan utama (*critical*), rangkaian kegiatan utama yang saling terhubung dinamakan jalur kritis (*critical path*). *Slack* atau *total float* dapat dihitung dengan menggunakan rumus:

$$TF = LS - ES \quad (5)$$

3. ANALISIS KEBUTUHAN

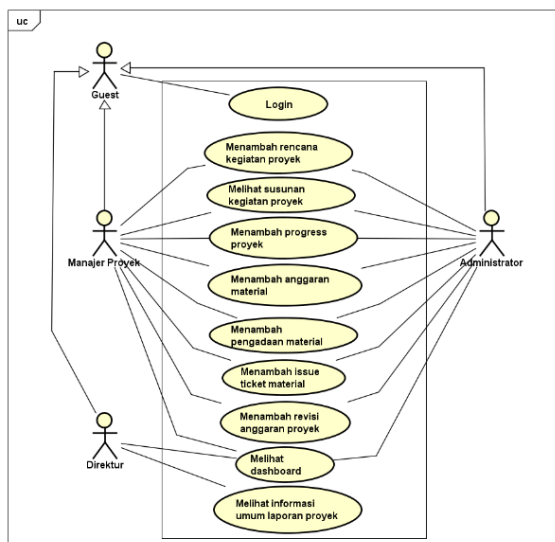
Tahap analisis kebutuhan pada penelitian ini diawali dengan elisitasi kebutuhan fungsional dan non fungsional perangkat lunak, identifikasi aktor yang terlibat dalam sistem, pendefinisian daftar spesifikasi kebutuhan yang nantinya ditranslasikan ke dalam diagram *use case*.

Tahapan ini dilakukan untuk menggambarkan kebutuhan-kebutuhan yang dimiliki oleh sistem. Dalam sistem aplikasi manajemen proyek terdapat 4 aktor, yaitu *guest*, *administrator*, direktur, dan manajer proyek. Kebutuhan dibuat berdasarkan dokumen laporan proyek perusahaan serta diintegrasikan dengan algoritma CPM untuk tahap perencanaan proyek. Untuk mengetahui deskripsi aktor lebih jelas dapat dilihat pada Tabel 1.

Tabel 1. Identifikasi aktor

Aktor	Deskripsi
Guest	Pengguna yang dapat melihat halaman awal sistem (halaman login) dan melakukan proses autentikasi login ke sistem.
Administrator	Pengguna yang memiliki otorisasi untuk dapat mengakses sebagian besar fitur aplikasi.
Direktur	Pengguna yang memiliki otorisasi untuk dapat melihat informasi mengenai laporan proyek dan melihat notifikasi adanya revisi anggaran pada proyek
Manajer Proyek	Pengguna yang memiliki otorisasi untuk dapat mengelola proyek yang dimilikinya

Kebutuhan fungsional pada perangkat lunak ini berjumlah 37 dan non fungsional berjumlah 1 yang juga dibuat berdasarkan dokumen laporan proyek perusahaan serta mengadaptasi hasil penelitian sebelumnya. Kebutuhan fungsional didetailkan kedalam fitur perangkat lunak berjumlah 104 buah. Berikut adalah sebagian *use case diagram* utama yang dapat dilihat pada Gambar 2.



Gambar 2. Use case diagram

4. PERANCANGAN

Perancangan perangkat lunak memuat hasil rancangan dari sistem yang dikembangkan, bertujuan untuk memberikan panduan dalam pembuatan perangkat lunak. Perancangan dilakukan berdasarkan analisis kebutuhan yang telah dibuat. Proses perancangan perangkat lunak pada penelitian ini terdiri dari pembuatan *sequence diagram*, pembuatan *class diagram* perancangan algoritma, perancangan basis data dan perancangan antarmuka

4.1 Perancangan Sequence Diagram

Sequence diagram (diagram urutan) yaitu diagram yang memperlihatkan atau menampilkan hubungan interaksi antar objek didalam sistem yang disusun berdasarkan urutan atau rangkaian waktu. Interaksi antar objek tersebut meliputi aktor, *display*, dan berupa pesan/*message* (Rosa & Shalahuddin, 2011).

Terdapat 5 *sequence diagram* yang ada pada penelitian ini diantaranya adalah menambah rencana kegiatan, merubah rencana kegiatan, menghapus rencana kegiatan, validasi rencana kegiatan, dan melihat susunan kegiatan.

4.2 Perancangan Class Diagram

Class diagram merepresentasikan struktur dari sistem dengan cara mendeskripsikan beberapa *class* yang akan dibangun pada sistem. Masing-masing *class* memiliki atribut, operasi atau method (Rosa dan Shalahuddin, 2011).

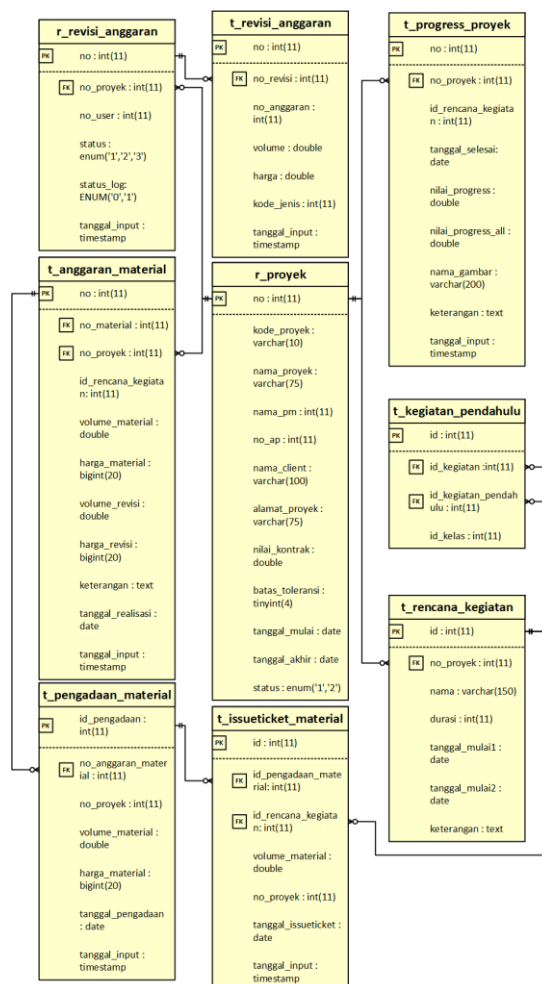
Class diagram pada penelitian ini dibagi kedalam 2 jenis yaitu *controller* dan *model*,

dengan rincian terdapat 12 *class* pada *controller* dan 12 *class* pada *model*.

4.3 Perancangan Data Model

Physical data model (PDM) digunakan dalam merancang struktur dari database sistem, Terdiri dari sekumpulan tabel yang saling berelasi. Pada PDM terdapat entitas yang menggambarkan nama tabel, atribut, relasi dan tipe datanya. Dengan penggunaan *database* dapat memudahkan proses penyimpanan dan integrasi data pada sistem.

Terdapat 28 struktur tabel pada perancangan data model sistem aplikasi manajemen proyek ini. Berikut adalah 9 struktur tabel beserta relasi antar tabel yang dapat mewakili keseluruhan struktur data model yang akan ditampilkan pada Gambar 3.



Gambar 3. PDM sistem aplikasi manajemen proyek

4.4 Perancangan Algoritma

Perancangan algoritma digunakan sebagai panduan dalam mengimplementasikan kode

program. Tabel 2 merupakan algoritma menentukan susunan kegiatan, Algoritma ini akan diimplementasikan pada fungsi `U_validasi_rencana()` yang terdapat pada class `M_proyek`. Ketika *administrator* atau manajer proyek mengakses fungsi ini maka sistem akan melakukan pengecekan status masing-masing kegiatan yang ada pada proyek tertentu apakah telah memenuhi persyaratan untuk dilakukan perhitungan CPM, contohnya seperti jumlah kegiatan paling awal dan paling akhir pada suatu proyek harus berjumlah maksimal 1 kegiatan. Selain itu sistem juga akan melakukan perhitungan estimasi tanggal pengerjaan masing-masing kegiatan berdasarkan durasi, hari libur, dan keterkaitan antara kegiatan satu dengan yang lainnya. Hasil akhir dari fungsi ini adalah status validasi proyek sudah tervalidasi atau belum.

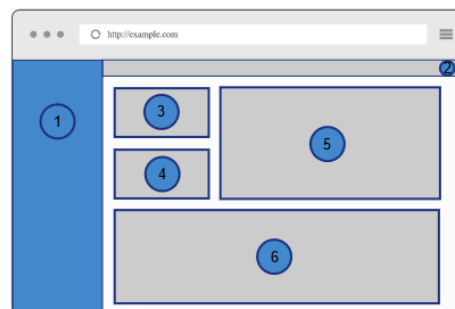
Tabel 2. Algoritma fungsi `U_validasi_rencana`

Baris	Pseudocode
1	Mulai
2	Menghitung jumlah rencana kegiatan yang ada pada database
3	Menghitung jumlah data dari hasil perhitungan fungsi <i>late finish</i>
4	Jika jumlah rencana kegiatan dan jumlah data <i>late finish</i> sama
5	Jika nilai <i>early start</i> dan <i>late start</i> sama
6	Jika terdapat hari libur pada proyek tersebut
7	Menghitung tanggal <i>early start</i> masing-masing kegiatan pada proyek tertentu berdasarkan urutan kegiatan dan libur pada hari minggu
8	Lainnya
9	Menghitung tanggal <i>early start</i> masing-masing kegiatan pada proyek tertentu berdasarkan urutan kegiatan tanpa menambahkan hari libur
10	Lainnya
11	Jika terdapat hari libur pada proyek tersebut
12	Menghitung tanggal <i>early start</i> dan <i>late start</i> masing-masing kegiatan pada proyek tertentu berdasarkan urutan

	kegiatan dan libur pada hari minggu
13	Lainnya
14	Menghitung tanggal <i>early start</i> dan <i>late start</i> masing-masing kegiatan pada proyek tertentu berdasarkan urutan kegiatan tanpa menambahkan hari libur
15	Menyimpan status validasi proyek pada database
16	Mengembalikan nilai variabel status validasi proyek berhasil
17	Lainnya
18	Mengembalikan nilai variabel status validasi proyek gagal
19	Selesai

4.5 Perancangan Antarmuka

Perancangan antarmuka digunakan sebagai panduan dalam mengimplementasikan antarmuka perangkat lunak. Dalam bagian ini akan dijelaskan gambaran umum mengenai desain antarmuka sistem aplikasi manajemen proyek beserta keterangannya. Gambar 4 menampilkan perancangan antarmuka halaman rencana kegiatan proyek.



Gambar 4. Perancangan antarmuka rencana kegiatan proyek

Perancangan antarmuka halaman rencana kegiatan berisi informasi mengenai daftar kegiatan pada suatu proyek beserta grafik CPM. Keterangan tentang halaman ini dijelaskan pada Tabel 3.

Tabel 3. Keterangan perancangan antarmuka rencana kegiatan proyek

No	Keterangan
1	Bagian ini berisi foto profil akun beserta nama usernya dan menu sidebar untuk melakukan navigasi ke halaman lain

2	Bagian ini berisi foto profil akun beserta nama usernya, menu topbar, dan terdapat tombol logout untuk keluar dari sistem
3	Bagian ini berisi combo box untuk memilih proyek yang diinginkan
4	Bagian ini berisi informasi rangkuman kegiatan proyek
5	Bagian ini berisi daftar kegiatan yang telah dimasukkan kedalam sistem
6	Bagian ini berisi grafik susunan kegiatan yang dibuat dengan berdasarkan CPM

5. IMPLEMENTASI

Pengembangan perangkat lunak sistem aplikasi manajemen proyek menggunakan bahasa pemrograman PHP sebagai pemroses logika aplikasi di sisi server dan *javascript* sebagai pemroses logika aplikasi di sisi *client*. Pengkodean PHP dibantu dengan menggunakan kerangka kerja CodeIgniter, dan pengkodean *javascript* dibantu dengan menggunakan *library* jQuery. Sistem manajemen basis data yang digunakan adalah MariaDB.

5.1 Implementasi Algoritma

Implementasi algoritma dibuat berdasarkan perancangan algoritma yang telah dirancang sebelumnya. Gambar 5 menunjukkan potongan kode program *U_validasi_rencana* pada kelas *M_proyek*.

1	<code>function U_validasi_rencana(\$id_proyek,</code>
2	<code>\$proyek, \$urutan) {</code>
3	
4	<code>\$count_rencana = count(\$this-</code>
5	<code>>get_rencana_kegiatan(\$id_proyek));</code>
6	<code>\$count_urutan = count(\$urutan);</code>
7	
8	
9	<code>if (\$count_rencana == \$count_urutan) {</code>
10	<code>echo \$proyek[0]['hari_libur'];</code>
11	<code>foreach (\$urutan as \$key => \$dt) {</code>
12	<code>if (\$dt['es'] == \$dt['ls']) {</code>
13	<code>if (\$proyek[0]['hari_libur'] == 1) {</code>
14	<code>\$temp_durasi = '+' . (\$dt['es'] -</code>
15	<code>1) . ' day';</code>
16	<code>\$update_tanggal = date("Y-m-d",</code>
17	<code>strtotime(\$temp_durasi,</code>
18	<code>strtotime(\$proyek[0]['tanggal_mulai']));</code>
19	<code>\$hitung_minggu = \$this-</code>
20	<code>>count_minggu(\$proyek[0]['tanggal_mulai'],</code>
21	<code>\$update_tanggal);</code>
22	<code>\$temp_durasi2 = '+' .</code>
23	<code>\$hitung_minggu . ' day';</code>
24	<code>\$update_tanggal2 = date("Y-m-d",</code>
25	<code>strtotime(\$temp_durasi2,</code>
26	<code>strtotime(\$update_tanggal));</code>
27	
28	
29	
30	<code>\$hitung_minggu = \$this-</code>
31	<code>>count_minggu(\$proyek[0]['tanggal_mulai'],</code>
32	<code>\$update_tanggal2);</code>
33	<code>\$temp_durasi3 = '+' , \$hitung_minggu</code>
34	<code>. ' day';</code>
35	
36	
37	

38	<code>\$tanggal_final = date("Y-m-d",</code>
39	<code>strtotime(\$temp_durasi3,</code>
40	<code>strtotime(\$update_tanggal));</code>
41	<code>} else {</code>
	<code>\$temp_durasi = '+' . (\$dt['es'] -</code>
	<code>1) . ' day';</code>
	<code>} }</code>

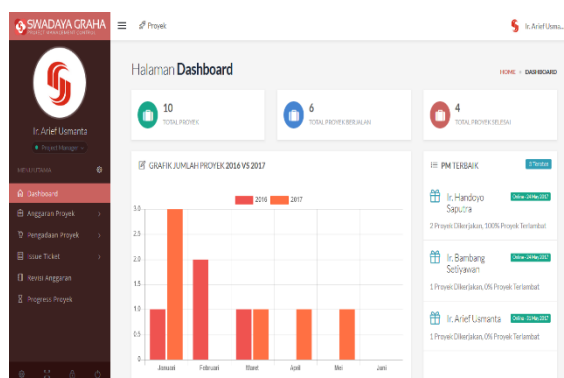
Gambar 5. Implementasi fungsi *U_validasi_rencana()*

5.2 Implementasi Basis Data

Pengimplementasian basis data dilakukan berdasarkan perancangan yang telah dibuat sebelumnya. Sistem aplikasi manajemen proyek menggunakan MariaDB sebagai tempat penyimpanan data perangkat lunak. Terdapat 28 struktur tabel pada implementasi basis data.

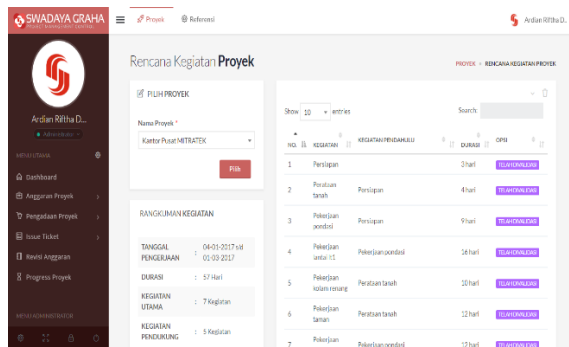
5.3 Implementasi Antarmuka

Implementasi antarmuka dibuat berdasarkan hasil perancangan antarmuka, desain tampilan antarmuka pada sistem aplikasi manajemen proyek dibuat dengan HTML dan CSS yang memanfaatkan kerangka kerja Bootstrap. Pada Gambar 6 menunjukkan implementasi antarmuka halaman *dashboard* dari aktor manajer proyek. Halaman dashboard adalah halaman yang ditampilkan pada saat *guest* berhasil melakukan login.

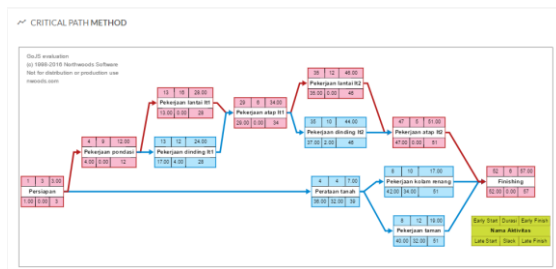


Gambar 6. Implementasi antarmuka *dashboard*

Implementasi antarmuka halaman rencana kegiatan ditunjukkan pada Gambar 7 dan Gambar 8. Rencana kegiatan proyek merupakan halaman yang ditampilkan setelah pengguna memilih menu proyek > rencana kegiatan proyek yang terdapat pada bagian atas (*topbar*) aplikasi. Pada halaman ini terdapat informasi mengenai rencana kegiatan proyek yang dibuat dalam bentuk Tabel dan grafik CPM.

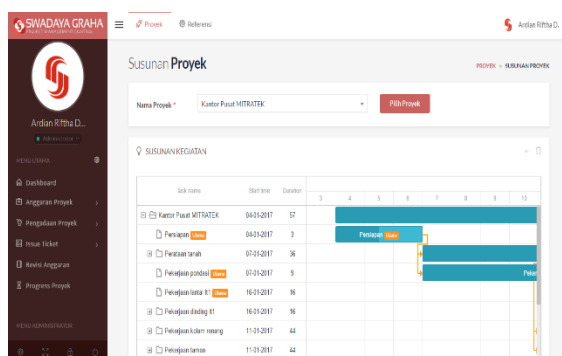


Gambar 7. Implementasi antarmuka rencana kegiatan proyek



Gambar 8. Implementasi antarmuka CPM

Implementasi antarmuka susunan kegiatan proyek ditunjukkan pada Gambar 9. Antarmuka susunan kegiatan proyek merupakan halaman yang ditampilkan setelah pengguna memilih menu proyek > susunan kegiatan proyek yang terdapat pada topbar aplikasi. Pada halaman ini terdapat informasi mengenai susunan kegiatan proyek yang dibuat dalam bentuk *gant* chart.



Gambar 9. Implementasi antarmuka susunan kegiatan proyek

6. PENGUJIAN

Pengujian bertujuan untuk memastikan bahwa perangkat lunak yang dibangun telah memenuhi spesifikasi yang telah ditentukan. Penelitian ini melibatkan 4 jenis pengujian, diantaranya adalah pengujian unit, pengujian integrasi, pengujian validasi dan pengujian kompatibilitas.

6.1 Pengujian Unit

Pengujian unit memiliki fokus untuk memverifikasi unit terkecil dari sebuah desain perangkat lunak yaitu komponen atau modul dari sebuah software. Dengan menggunakan desain level komponen, dapat digunakan sebagai panduan untuk menguji jalur kontrol yang penting untuk menunjukkan kesalahan yang terdapat pada suatu modul. (Pressman, 2010).

Hasil pengujian unit pada penelitian ini menunjukkan bahwa pengujian unit fungsi menentukan tanggal dan status kegiatan mendapatkan nilai *cyclomatic complexity* 5, fungsi menentukan susunan kegiatan mendapatkan nilai *cyclomatic complexity* 8, dan fungsi menampilkan detail laporan proyek mendapatkan nilai *cyclomatic complexity* 7. Nilai *cyclomatic complexity* memiliki arti pada kode, algoritma, atau program dimana nilai 1-10 menandakan bahwa sistem mudah dipahami, mudah diimplementasikan, mudah diperbaiki, mudah diuji serta resiko terjadinya kesalahan sistem relatif rendah (Institute, 1997).

6.2 Pengujian Integrasi

Pengujian integrasi memiliki fokus untuk menguji gabungan fungsi komponen perangkat lunak, perangkat keras, ataupun keduanya telah bekerja sama dengan benar.

Penelitian ini menguji 2 *class* yang saling berhubungan yaitu *class C_progress_proyek* dan *M_progress_proyek*. Hasil pengujian integrasi pada sistem aplikasi manajemen proyek menunjukkan bahwa kedua *class* dapat bekerja sama dengan baik. masukan dari fungsi *C_progress_proyek* dapat diproses oleh *M_progress_proyek* sehingga menghasilkan keluaran seperti yang diharapkan.

6.3 Pengujian Validasi

Pengujian validasi digunakan untuk menganalisa apakah sistem yang dibangun telah sesuai dengan yang dibutuhkan. Komponen-komponen yang telah dirumuskan dalam daftar kebutuhan perangkat lunak akan menjadi acuan dalam melakukan pengujian validasi.

Tabel 4 menunjukkan hasil pengujian validasi menambah data *progress* proyek yang berfungsi untuk memasukkan *progress* kegiatan tertentu yang di laporkan secara rutin (harian, mingguan, dan bulanan) kepada perusahaan.

Tabel 4 Kasus uji menambah data *progress* proyek

Nama Kasus Uji	Menambah Data Progress Proyek
Obyek Pengujian	F32-SRS-F-02
Tujuan Pengujian	Untuk memastikan bahwa sistem dapat menambah data <i>progress</i> proyek.
Prosedur Pengujian	<ol style="list-style-type: none"> 1. Aktor memilih menu <i>Progress</i> Proyek. 2. Aktor memilih proyek 3. Aktor memilih tombol Tambah <i>Progress</i>. 4. Aktor mengisi formulir input <i>progress</i> proyek 5. Aktor memilih tombol simpan.
Hasil Yang Diinginkan	Sistem berhasil menyimpan data <i>progress</i> proyek kedalam sistem basis data.
Hasil Pengujian	Sistem menyimpan data <i>progress</i> proyek kedalam sistem basis data dan menampilkan notifikasi bahwa data berhasil disimpan
Status	Valid.

6.4 Pengujian Kompatibilitas

Pengujian kompatibilitas memiliki fokus pada persyaratan non-fungsional, pengujian ini memiliki tujuan untuk memastikan perangkat lunak yang telah dibangun mampu berjalan dengan baik pada lingkungan yang heterogen, pengujian kompatibilitas juga melibatkan pemilihan konfigurasi lingkungan tertentu (Yoon, et al., 2008).

Pengujian kompatibilitas yang diterapkan pada sistem ini dilakukan pada sisi perambah web. Pengujian kompatibilitas dilakukan dengan menggunakan perangkat lunak bernama SortSite versi 5.25.784.0. Alat ini menganalisa struktur *website* untuk mengetahui apakah terdapat bagian yang tidak kompatibel dengan perambah web. Pada Tabel 5 menunjukkan macam-macam perambah web yang digunakan SortSite dalam pengujian kompatibilitas.

Tabel 5 Browser yang digunakan pada pengujian kompatibilitas

No	Nama Browser	Versi Browser
1	Internet Explorer	9, 10, dan 11
2	Edge	15
3	Firefox	<= 53
4	Safari	<= 10
5	Opera	44
6	Chrome	58
7	iOS	<= 10
8	Android	<=4

Pengujian dimulai dengan cara memasukkan alamat *website* pada kolom url yang tersedia pada SortSite, setelah itu menekan tombol *Start Check* untuk melakukan pengecekan pada seluruh halaman yang ada di *website*. Gambar 10 merupakan hasil pengujian kompatibilitas dengan menggunakan perangkat lunak SortSite.

Browser	IE			Edge		Firefox		Safari		Opera	Chrome		iOS			Android	
Version	9	10	11	15	≤ 52	53	≤ 9	10	44	58	≤ 8	9	10	≤ 3	4*		
Critical Issues	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
Major Issues	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
Minor Issues	⚠	⚠	⚠	✓	⚠	⚠	⚠	⚠	✓	⚠	✓	✓	✓	✓	✓		

Gambar 10. Implementasi antarmuka susunan kegiatan proyek

Hasil dari pengujian kompatibilitas pada Gambar 7.4 dapat membantu mengetahui apakah terdapat masalah dalam sistem jika digunakan di berbagai perambah web. Terdapat tiga kategori masalah yang ada didalam pengujian kompatibilitas yaitu *critical issues*, *major issues*, dan *minor issues*. *Critical issues* mengindikasikan sebuah konten atau fitur yang tidak didukung pada beberapa perambah web. *Major issues* mengindikasikan masalah utama pada tampilan atau performa pada beberapa perambah web. *Minor issues* mengindikasikan masalah minor pada tampilan atau performa pada beberapa perambah web.

7. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, maka dapat diambil kesimpulan sebagai berikut:

1. Pada tahap perancangan sistem menghasilkan *class diagram*, *sequence diagram*, *physical data model*, perancangan komponen, dan perancangan antarmuka. *Class diagram* yang dihasilkan dibedakan kedalam 2 *class diagram* yaitu *class controller* yang terdiri dari 12 *class*, dan *class model* yang juga terdiri dari 12 *class*. *Sequence diagram* yang dihasilkan dan didokumentasikan dalam dokumen ini adalah 5 *sequence diagram* yang dapat mewakili 5 fungsional dasar pada aplikasi. Pada perancangan komponen hanya mencantumkan algoritma untuk menentukan tanggal dan status kegiatan, menentukan susunan kegiatan, dan menentukan detail laporan peritem. Perancangan sistem digunakan sebagai

- dasar dalam melakukan implementasi sistem.
2. Proses implementasi sistem dilakukan berdasarkan perancangan sistem yang telah didefinisikan. Critical path method sebagai metode utama dalam penelitian ini dapat digunakan dalam membuat susunan kegiatan proyek, sebagai acuan dalam menentukan tanggal dimulainya dan berakhirnya kegiatan, sebagai acuan dalam merelasikan antara kegiatan dengan biaya, serta dapat digunakan sebagai acuan untuk memantau keterlambatan progress proyek. Fungsi pengelolaan anggaran, pengadaan, dan *issue ticket* dapat digunakan manajer proyek dalam mengelola biaya proyek. Sedangkan fungsi lain seperti melihat laporan proyek dapat digunakan direktur untuk mengawasi dan sebagai acuan dalam membuat keputusan yang terkait dengan proyek.
 3. Proses pengujian dilakukan dengan menggunakan pengujian unit, pengujian validasi, dan pengujian kompatibilitas. Berdasarkan hasil pengujian unit, 3 fungsi yang diuji berada pada kategori 1-10, dalam arti fungsi yang dibuat mudah untuk dipahami, diuji serta diimplementasikan atau diperbaiki. Berdasarkan hasil *validation testing*, fungsi yang ada di dalam sistem aplikasi manajemen proyek berjalan dengan baik, dibuktikan dengan status valid yang terdapat pada 68 fungsional sistem yang diuji dari jumlah keseluruhan fungsional sistem yaitu 104 fungsional. Berdasarkan hasil Pengujian kompatibilitas, sistem aplikasi manajemen proyek dapat digunakan dengan baik pada 8 macam jenis perambah web.
- 01.pdf > [Diakses 26 januari 2017].
- Mubarak, S., 2010. Construction Project Scheduling and Control Second Edition. Hoboken: John Wiley & Sons, Inc. Tersedia di <
<http://en.bookfi.net/book/1186772>>
 [Diakses 21 november 2016].
- Pressman, Roger S., 2010. Software Engineering A Practitioner's Approach. 7th ed. New York: McGraw-Hill.
- Rosa A.S. dan Salahuddin M., 2011. Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek). Bandung: Modula.
- Yoon, C. I., Sussman, A. & Porter, A., 2008. Effective and Scalable Software Compatibility Testing. College Park, MD, USA: University of Maryland.

DAFTAR PUSTAKA

- Heizer, J. dan Render, B., 2006. Operation Management. New Jersey: Pearson.
- Institute, P.M., 2008. A Guide To The Project Management Body of Knowledge. [pdf] Pennsylvania: Project Management Institute. Tersedia di:
 <<http://en.bookfi.net/book/1023737>>
 [Diakses 26 januari 2017].
- Institute, S.E., 1997. C4 Software Technology Reference Guide. [pdf] Software Engineering Institute. Tersedia di:
 <<http://www.sei.cmu.edu/reports/97hb0>