

PENGEMBANGAN APLIKASI PENJADWALAN RAPAT MENGUNAKAN METODE *PHASED DEVELOPMENT*

DEVELOPMENT OF MEETING SCHEDULE APPLICATION USING PHASED DEVELOPMENT METHOD

Egia Rosi Subhiyakto*¹, Yani Parti Astuti²

*Email: egia@dsn.dinus.ac.id

^{1,2}Program Studi S1 Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro, Semarang

Abstrak— Penggunaan teknologi informasi di sebuah institusi menjadi hal yang penting dalam mendukung kegiatan rutin sehari-hari, salah satunya adalah agenda rapat. Terjadinya bentrokan jadwal maupun ruangan menjadi urgensi penelitian ini. Di samping itu, pencatatan rapat masih dilakukan dalam buku atau memo juga meningkatkan risiko kehilangan dokumen baik karena keteledoran atau bencana alam. Tujuan dari penelitian ini adalah memberikan alternatif solusi permasalahan tersebut dengan membangun aplikasi penjadwalan rapat menggunakan bahasa pemrograman PHP dan basis data MySQL. Analisis dilakukan menggunakan analisis berorientasi objek menggunakan beberapa diagram seperti *diagram use case*, diagram aktivitas dan diagram kelas. Metode pengembangan sistem yang digunakan adalah metode *phased development*. Metode ini terdiri dari beberapa tahapan yakni perencanaan, analisis, perancangan, implementasi lalu menghasilkan versi aplikasi. Berdasarkan hasil pengujian *black box* dari enam fitur utama menunjukkan bahwa aplikasi yang dibangun sudah memenuhi standar fungsionalitas yang diharapkan. Hasil *user acceptance test* kinerja sistem dapat dikatakan 100% sudah memenuhi harapan karena berdasarkan hasil evaluasi pengguna mereka sangat setuju dan setuju terhadap semua parameter yang dievaluasi. Keunggulan dari aplikasi yaitu kepuasan pengguna didapatkan 100% pengguna puas menggunakan aplikasi, antarmuka sistem menarik dan akan merekomendasikan aplikasi tersebut. Hanya saja untuk fungsionalitas sistem terdapat 7 responden yang menginginkan penambahan fitur baru.

Kata kunci — Jadwal, rapat, *phased development*, *unified modeling language*.

Abstract— The use of information technology in institutions is important to support daily routine activities, one of them is meeting. The occurrence of room and schedule conflict is the urgency of this research. Besides that, the recording of meetings is still carried out in memos also increases the risk of losing documents either due to negligence or natural disasters. The aim of this study is to provide an alternative solution to the problem by building a meeting scheduling application using the PHP and MySQL database. Analysis is carried out using object-oriented analysis using several diagrams such as use case diagrams, activity diagrams, and class diagrams. The system development method used is the phased development method. Based on the results of black box testing of the six main features, it shows that the application has met the expected standard of functionality. The results of user acceptance test system performance almost 100% fulfilling expectations because they are strongly agreed and agreed to all parameters evaluated. It is obtained from the application that is 100% of users are satisfied, the system interface is interesting and will recommend the application. However, there are 7 respondents who want new features for the system functionalities.

Keywords — Schedule, meeting, phased development, unified modeling language.

I. PENDAHULUAN

Informasi menjadi bagian yang fundamental sudah menjadi kebutuhan sehari-hari di era saat ini. Menurut data Badan Pusat Statistik (BPS) tahun 2017, terdapat peningkatan presentase penduduk yang mengakses internet pada tahun 2013 sekitar

14,9% menjadi 32,34% pada tahun 2017. Hal tersebut menjadi bukti bahwa akses informasi khususnya melalui internet semakin meningkat dari tahun ke tahun [1]. Hal tersebut merupakan salah satu faktor mempercepat berkembangnya teknologi informasi. Penggunaan teknologi informasi dewasa ini menjadi tren baik di organisasi publik maupun

swasta dalam rangka meningkatkan kinerja institusi tersebut. Tidak dipungkiri juga bahwa dengan adanya teknologi informasi dapat meningkatkan produktivitas pekerjaan dalam sebuah lembaga.

Institusi yang besar seperti institusi pemerintahan pastinya memiliki kegiatan yang rutin dan terjadwal. Salah satu contohnya adalah rapat yang sering diadakan dalam kegiatan sehari-hari. Informasi jadwal rapat mayoritas masih ditulis dalam sebuah buku besar atau memo sehingga kurang praktis. Faktor lain adalah penggunaan buku besar tidak efisien dalam mengatur jadwal rapat yang relatif banyak. Selain itu antisipasi jadwal rapat yang bentrok juga masih belum dapat diminimalisir. Saat mengatur pertemuan, peserta rapat harus mencapai kesepakatan bersama untuk mengadakan pertemuan yang tunduk pada kendala dan preferensi pribadi mereka. Ini adalah proses yang memakan waktu, dan berbagai aplikasi kalender digunakan membantu pengguna untuk menjadwalkan rapat.

Teknologi informasi dapat menjadi sebuah alternatif solusi dari masalah yang telah dijelaskan. Pengembangan sebuah aplikasi harus mengikuti tahapan pengembangan seperti proses analisis, perancangan, proses coding, pengujian dan pemeliharaan sistem. Untuk menjadi seorang pengembang perangkat lunak yang baik harus menguasai semua tahapan pengembangan [2]. Aplikasi perangkat lunak gagal mengatasi kendala proses penjadwalan tradisional dan berfungsi sebagai alat pendukung untuk mengelola informasi rapat. Salah satu kendala utama dalam penjadwalan otomatis adalah tidak adanya standar protokol komunikasi terstruktur. Selain itu, penjadwalan otomatis membutuhkan masalah lain yang perlu dipertimbangkan seperti paradigma pengambilan keputusan otomatis, mekanisme pemilihan strategi negosiasi, dan lain lain [3].

Penelitian ini membahas tentang pengembangan aplikasi untuk penjadwalan rapat dalam sebuah instansi dengan tujuan untuk mengatasi permasalahan penjadwalan rapat menjadi lebih terstruktur dan terkomputerisasi. Aplikasi yang dibangun dapat menjadi alternatif solusi proses penjadwalan rapat agar tidak terjadi bentrok, lebih tertata dan terkomputerisasi dengan baik dalam sebuah *database*.

II. TINJAUAN PUSTAKA

Penelitian terkait penjadwalan rapat yang telah dilakukan sebelumnya dilakukan oleh [4] [5] yang membahas tentang negosiasi otomatis dan *deadline*

penjadwalan. Hal tersebut merupakan salah satu masalah utama yang sedang berlangsung selama dua dekade terakhir. Dalam hasil penelitian tersebut diusulkan protokol negosiasi agen untuk memenuhi penjadwalan yang menjadi salah satu masalah terdistribusi alami yang menonjol, sehingga dapat diasumsikan bahwa ada agen penjadwalan dan memiliki pengetahuan tentang preferensi pengguna, profil peserta rapat, memegang mekanisme penalaran untuk mengevaluasi undangan rapat, dan mampu memilih strategi negosiasi secara otomatis. Protokol negosiasi yang diusulkan membantu agen penjadwalan pertemuan untuk menangani skenario negosiasi bilateral dan multilateral. Protokol membantu proses negosiasi otomatis dan efektivitasnya selama kegiatan penjadwalan.

Penelitian lain yang dilakukan dalam [6] memaparkan tentang penjadwalkan pertemuan yang melibatkan orang dengan komitmen dan preferensi yang berbeda merupakan tugas yang sulit. Penjadwalan rapat sulit karena aplikasi *online* saat ini tidak dapat menangani tanggung jawab slot waktu mengelola secara otomatis dan otonom. Dalam hasil penelitian tersebut disajikan arsitektur sistem multi-agen di mana setiap orang diwakili oleh agen. Agen-agen ini secara otomatis dan otonom bekerja bersama untuk membantu pengguna yang berbeda untuk memesan rapat atas nama mereka. Setiap agen memiliki kemampuan untuk mengelola, menegosiasikan dan menjadwalkan tugas, rapat, acara, janji untuk pengguna yang ditetapkan. Dalam sistem *multi-agen* ini, agen mengoordinasikan aktivitasnya dan bernegosiasi atas nama pengguna yang terkait untuk menemukan solusi yang memenuhi persyaratan dan preferensi rapat pengguna. Prototipe sistem ini diimplementasikan untuk menunjukkan bagaimana agen dapat secara otomatis memesan rapat.

Penelitian sebelumnya oleh Shakshuki E.M, penjadwalan otomatis membutuhkan masalah lain yang perlu dipertimbangkan seperti paradigma pengambilan keputusan otomatis, mekanisme pemilihan strategi negosiasi, dan lain-lain [3]. Penelitian tersebut mengusulkan agen penjadwalan pertemuan pribadi (*Personal Meeting Scheduling Agent* atau PMSA) dan protokol penjadwalan pertemuan pribadi (*Personal Meeting Scheduling Protocol* atau PMSP). PMSP tertanam dalam PMSA untuk menangani negosiasi bilateral dan multilateral. PMSA dirancang dengan menggunakan metodologi berbasis tujuan dan berbasis model. Selain itu, PMSP dirancang mengikuti protokol negosiasi terstruktur yang dipengaruhi oleh protokol respons

simultan. Untuk mengevaluasi semua undangan rapat dan untuk membuat keputusan berdasarkan preferensi pengguna, profil peserta, dan ketersediaan jadwal, penelitian tersebut menggunakan model *naïve Bayes* dari estimasi kemungkinan maksimum. Tujuan PMSP adalah untuk secara otomatis mengambil keputusan dan memilih strategi negosiasi yang tepat untuk menghindari atau menyelesaikan kemungkinan konflik rapat.

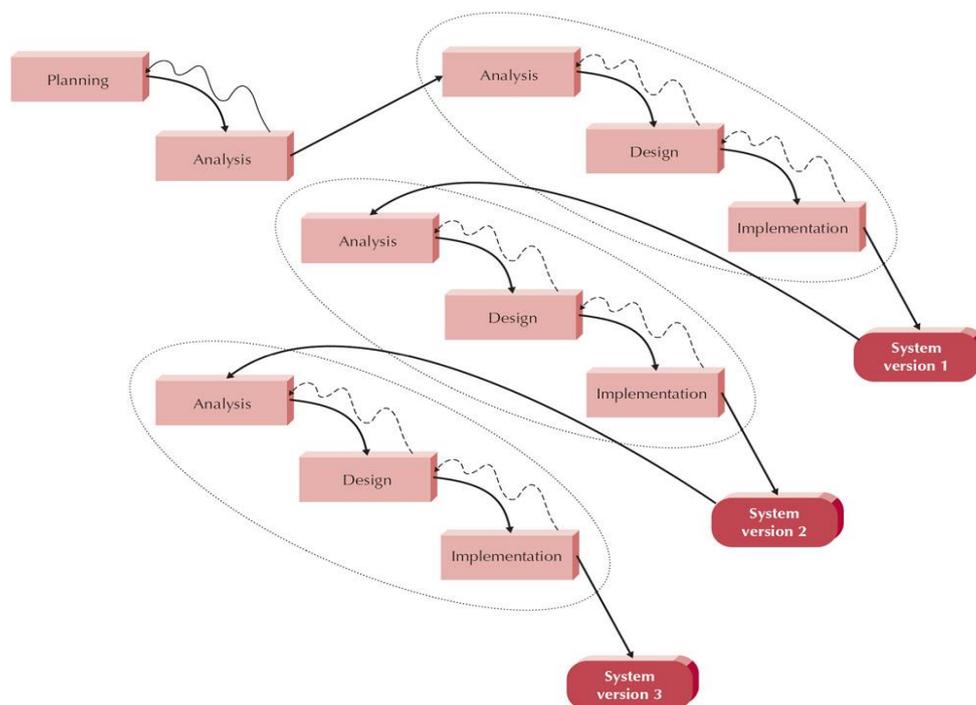
Penelitian yang terkait dengan penelitian ini adalah penelitian dengan menggunakan algoritma semut untuk menyelesaikan masalah penjadwalan mata kuliah dan dosen yang dilakukan secara rutin setiap semester [7]. Algoritma semut yang digunakan untuk penjadwalan menggunakan beberapa parameter yakni tingkat prioritas dosen, waktu, mata kuliah dan ruangan. Sedangkan yang terkait dengan *phased development* ada pada penelitian [8] yang membahas tentang platform metode *rapid application development* atau pengembangan cepat. Dalam penelitian tersebut dua *mode platform* pengembangan cepat diperkenalkan secara singkat, dan dianalisis kekurangannya. Penelitian tersebut mengusulkan kerangka berbasis *template* untuk *platform* pengembangan aplikasi cepat, dan berbagai komponen kerangka diperkenalkan. Pada akhirnya, *platform* dan metode pengembangan umum dibandingkan, hasilnya membuktikan bahwa *platform* yang diusulkan lebih

efisien daripada metode pengembangan reguler dalam laba pengembangan. Penelitian yang dilakukan dalam [9] menerapkan metode RAD (*Rapid Application Development*) untuk menghasilkan sistem penjualan sepeda *online* melalui integrasi basis data dan membangun hubungan dengan pelanggan secara lebih personal.

III. METODE

Metode pengembangan sistem yang digunakan dalam artikel ini adalah metode *phased development*. Metode ini merupakan salah satu jenis dari metode *Rapid Application Development* (RAD) selain metode *prototyping* dan metode *throw-away prototyping* [10]. Metode *phased development* menghasilkan versi aplikasi yang memiliki series. Keunggulan dari metode ini adalah menghasilkan sistem yang berguna bagi pengguna secara cepat dan banyak fungsi penting yang diuji dengan lebih baik. Terdapat beberapa penelitian yang menggunakan metode terkait seperti dalam penelitian [11][12]. Metode pengembangan sistem yang lain yakni *waterfall* [13] dan *prototyping* [14] juga *familiar* untuk digunakan.

Gambar-1 menggambarkan metode *phased development*. Penerapan metode *phased development* dalam artikel ini adalah sebagai berikut:



Gambar-1. Metode *phased development* [9]

A. *Planning*

Tahap pertama adalah *planning* atau perencanaan. Perencanaan dilakukan melalui proses pengumpulan data dari *stakeholder* terkait seperti instansi-instansi yang akan menggunakan aplikasi penjadwalan rapat. Jadwal pengembangan juga direncanakan mulai bulan Oktober sampai bulan Desember tahun 2018. Selain itu juga dilakukan pengumpulan kebutuhan-kebutuhan aplikasi yang dibangun.

B. Analisis

Tahap kedua adalah proses analisis. Proses analisis yang dilakukan adalah analisis kebutuhan-kebutuhan seperti pemodelan kebutuhan menggunakan diagram *Unified Modeling Language* (UML). Dalam artikel ini disajikan dalam diagram *use case* dan diagram aktivitas.

C. Perancangan

Pada fase perancangan dilakukan perancangan arsitektur, perancangan basis data dan perancangan antarmuka.

D. Implementasi

Tahapan ini melakukan proses koding sesuai dengan hasil analisis dan perancangan yang telah dilakukan sebelumnya. Implementasi yang dilakukan disesuaikan dengan versi aplikasi yang dibangun.

E. Versi sistem atau aplikasi

Aplikasi yang dihasilkan memiliki beberapa seri/versi yang berbeda sesuai tahapan pengembangan yang dilakukan. Seperti yang ditunjukkan pada gambar terdapat beberapa versi aplikasi yang dihasilkan.

Penelitian ini lebih fokus kepada *phased development* seri/versi yang pertama. Sehingga aplikasi yang dihasilkan merupakan aplikasi penjadwalan rapat versi pertama. Proses pengujian dan evaluasi aplikasi versi pertama akan dibahas dalam bagian selanjutnya.

IV. HASIL DAN PEMBAHASAN

Bagian ini membahas tahapan *phased development* yang merupakan metode pengembangan sistem yang digunakan. Dalam pembahasan ini fokus kepada seri/versi pertama dari sistem yang dibangun. Tahapan *planning* atau perencanaan pada metode *phased development* telah dilakukan dengan pengumpulan data dan kebutuhan dengan *stakeholder* terkait. Berikut adalah penjabaran lebih lengkap mengenai tahapannya.

A. Analisis

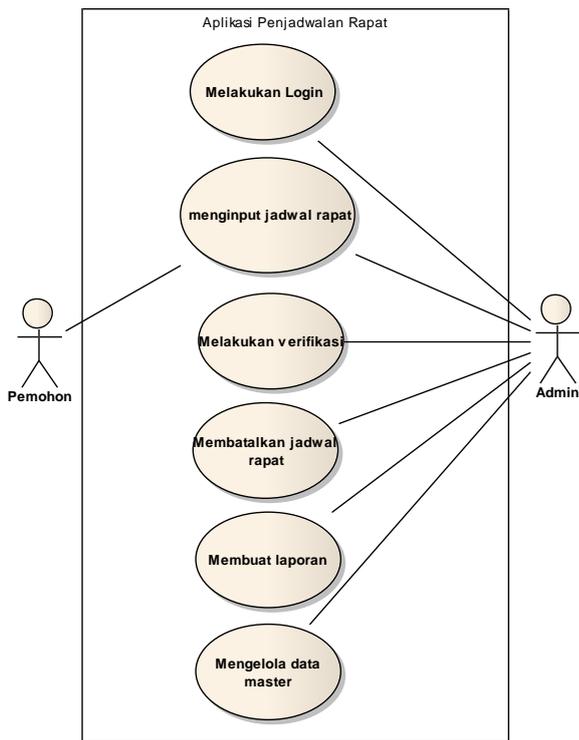
Tahapan analisis dilakukan setelah proses rekayasa kebutuhan. Analisis dilakukan untuk memodelkan masalah sehingga siap untuk didesain solusinya. Tahapan analisis dilakukan dengan melakukan pemodelan sistem menggunakan UML diagram (*Unified Modeling Language*). Tabel-1 menunjukkan daftar kebutuhan fungsional aplikasi yang akan dibangun.

Tabel-1. Kebutuhan fungsional penjadwalan rapat

No	Kebutuhan Fungsional	Keterangan
1.	Melakukan <i>Login</i>	Digunakan oleh pemohon maupun admin untuk mengakses sistem
2.	Menginput Jadwal Rapat	Digunakan oleh pemohon maupun admin untuk melakukan <i>input</i> jadwal rapat
3.	Verifikasi admin	Dilakukan oleh admin untuk verifikasi jadwal rapat
4.	Membatalkan Jadwal	Dilakukan oleh admin untuk melakukan pembatalan jadwal rapat
5.	Membuat Laporan	Pencetakan laporan dilakukan oleh admin untuk pelaporan rekapitulasi jadwal rapat
6.	Pengolahan Data Master	Terdapat beberapa data master seperti data bidang, data ruangan, data data waktu rapat yang dikelola oleh admin.

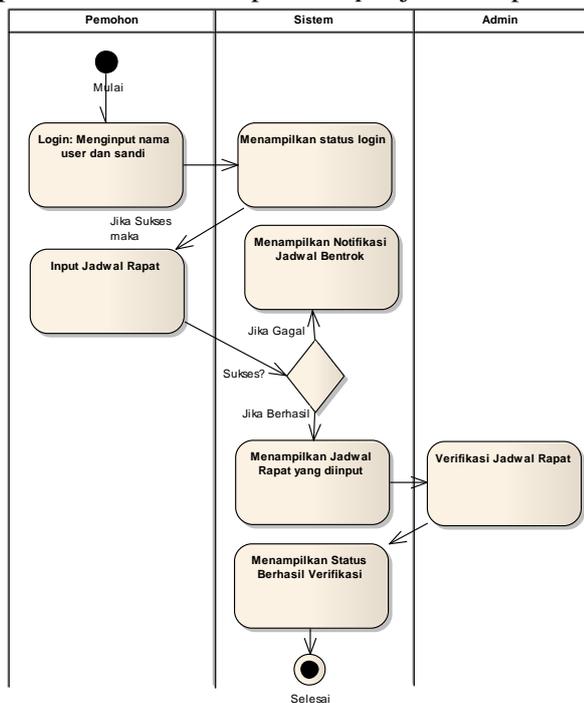
Pemodelan untuk menggambarkan fungsionalitas sistem dapat menggunakan diagram *use case*. Penggambaran diagram *use case* merujuk pada Tabel-1 kebutuhan fungsional, pada tabel tersebut terdapat dua aktor yakni pemohon dan *admin*. Identifikasi *use case* juga perlu dilakukan antara lain menentukan beberapa kandidat *use case* yakni melakukan *login*, menginput jadwal rapat, verifikasi admin, membatalkan jadwal rapat, membuat laporan dan mengelola data master.

Gambar-2 menunjukkan diagram *use case* penjadwalan rapat. Pemodelan seperti sudah dijelaskan sebelumnya, berdasarkan dari daftar kebutuhan fungsional hasil analisis. Kedua aktor memiliki hak akses yang berbeda terhadap sistem. Jika pemohon hanya dapat melakukan *login* dan menginput jadwal rapat, maka admin dapat menggunakan semua fitur di dalam sistem. Sedangkan jalannya aplikasi digambarkan dalam sebuah diagram aktivitas, seperti yang tergambar dalam Gambar-3.



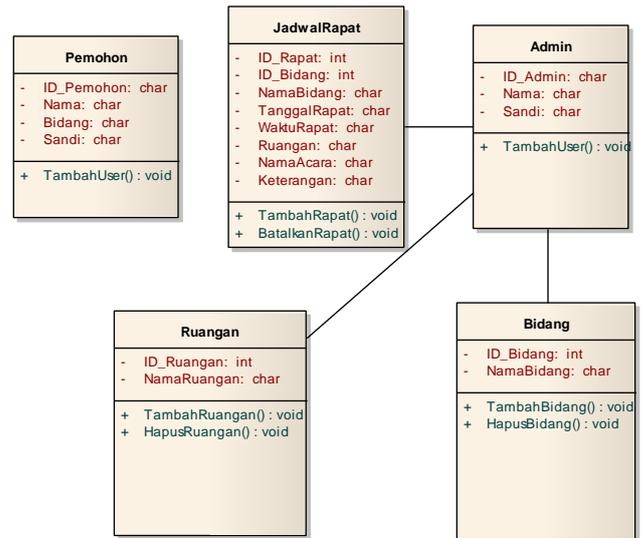
Gambar-2. Diagram use case penjadwalan rapat

Diagram aktivitas terdiri dari 3 bagian *swimlane* yakni pemohon, sistem dan *admin*. Pertama, pemohon melakukan *login* dengan menginput nama *user* dan sandi. Apabila berhasil maka sistem akan menampilkan status *login* berhasil, kemudian pemohon melakukan proses *input* jadwal rapat.



Gambar-3. Diagram activity penjadwalan rapat

Terdapat dua kemungkinan, yang pertama apabila gagal dalam artian jadwal rapat bentrok dengan jadwal lain baik dari sisi waktu maupun ruangan maka sistem akan menampilkan notifikasi bahwa jadwal bentrok. Apabila input berhasil maka sistem akan menampilkan detail jadwal rapat yang sudah diinput. Admin kemudian akan melakukan proses verifikasi jadwal rapat untuk selanjutnya ditampilkan oleh sistem.

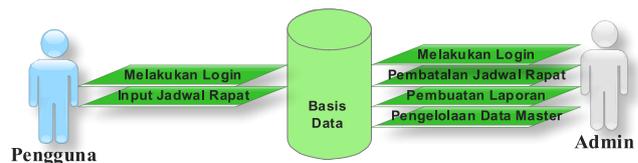


Gambar-4. Diagram kelas penjadwalan rapat

Pemodelan kelas-kelas dan hubungannya digambarkan dengan diagram kelas seperti yang ditunjukkan dalam Gambar-4. Terdapat lima kelas utama yakni kelas pemohon, kelas jadwal rapat, kelas admin, kelas ruangan dan kelas bidang yang memiliki atribut dan *method* masing masing. Kelas pemohon dan kelas admin memiliki relasi terhadap kelas jadwal rapat. Sedangkan kelas ruangan dan kelas bidang memiliki relasi terhadap kelas admin.

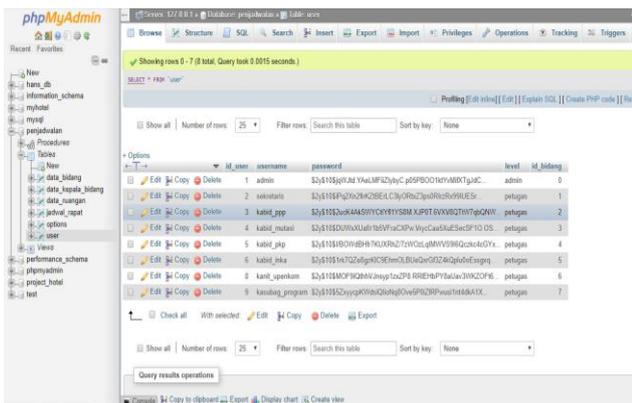
B. Perancangan

Tahap perancangan dilakukan untuk memodelkan solusi agar siap diimplementasikan dalam bentuk koding. Yang dilakukan meliputi perancangan arsitektur sistem, perancangan basis data, dan perancangan antarmuka aplikasi yang akan dibangun.



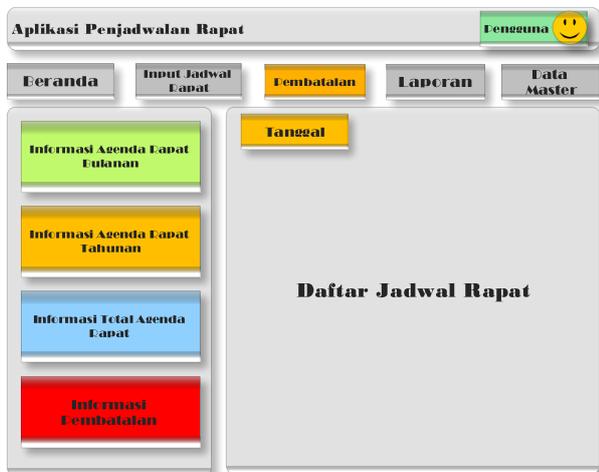
Gambar-5. Arsitektur aplikasi

Gambar-5 menunjukkan arsitektur aplikasi penjadwalan rapat yang dibangun. Terdapat beberapa bagian yakni pengguna, admin, fitur aplikasi dan basis data aplikasi. Berdasarkan arsitektur tersebut pengguna aplikasi dapat melakukan *login*, menginput jadwal rapat kemudian akan disimpan dalam sebuah basis data. Sedangkan admin dapat melakukan login, melakukan pembatalan jadwal rapat, pembuatan laporan, dan pengelolaan data master.



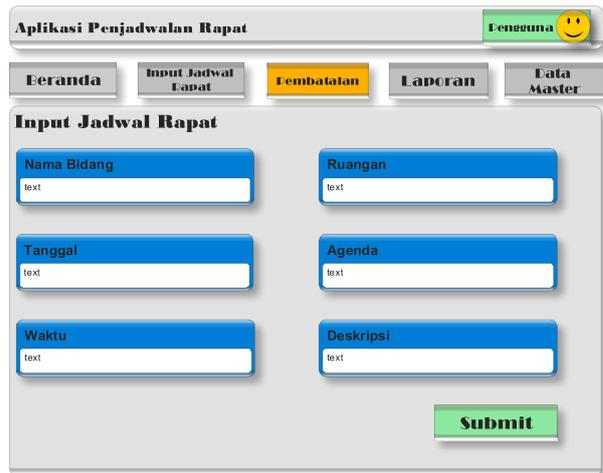
Gambar-6. Perancangan basis data

Gambar-6 menunjukkan perancangan basis data aplikasi yang dibangun. Terdapat beberapa tabel dalam basis data tersebut yakni tabel data bidang, tabel data kepala bidang, tabel data ruangan, tabel jadwal rapat, tabel pilihan, dan tabel user. Selanjutnya dilakukan perancangan antarmuka aplikasi yang dibangun. Perancangan yang ditampilkan antara lain perancangan halaman utama aplikasi, halaman *input* jadwal rapat dan perancangan halaman pembatalan rapat.



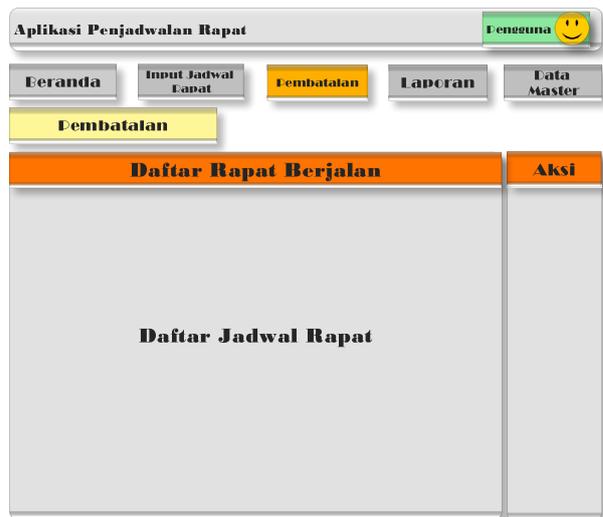
Gambar-7. Perancangan halaman utama

Gambar-7 menunjukkan halaman utama aplikasi yang mana terdapat menu akses seperti menu *input*, menu pembatalan, menu pencetakan laporan dan menu pengolahan data master. Selain itu juga terdapat beberapa informasi seperti informasi agenda rapat bulanan, informasi agenda rapat tahunan, informasi total agenda rapat dan informasi pembatalan.



Gambar-8. Perancangan halaman *input* jadwal

Gambar-8 menunjukkan halaman *input* jadwal rapat yang mana terdapat beberapa isian yakni nama bidang, tanggal rapat, waktu atau jam rapat, ruangan rapat, agenda rapat dan deskripsi rapat.



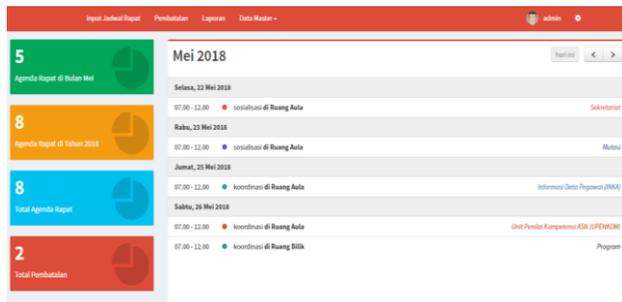
Gambar-9. Perancangan halaman pembatalan

Gambar-9 menunjukkan halaman pembatalan rapat yang mana terdapat daftar jadwal rapat dan aksi pembatalan yang akan dilakukan. Aksi pembatalan akan dilakukan oleh admin aplikasi

apabila terdapat kejadian mendadak atau hal-hal yang menyebabkan pembatalan dapat dilakukan.

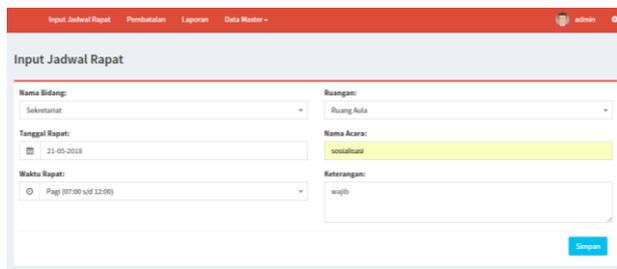
C. Implementasi

Tahap analisis dan perancangan telah dilakukan, pada bagian ini akan dibahas mengenai implementasi aplikasi yang dilakukan. Implementasi pembangunan aplikasi dilakukan menggunakan Bahasa pemrograman PHP, HTML, dan *JavaScript*. Sedangkan untuk basis data menggunakan MySQL dengan bantuan PHP *Myadmin* untuk membuat basis datanya.



Gambar-10. Halaman utama

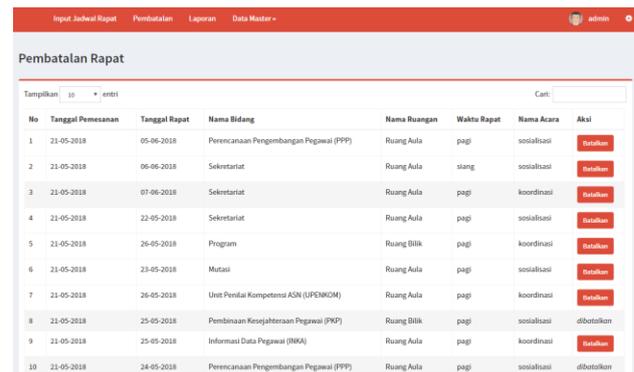
Gambar-10 menunjukkan halaman utama dari aplikasi yang dibangun. Pada bagian kiri, terdapat beberapa informasi mengenai agenda rapat bulan berjalan, informasi agenda rapat tahun berjalan, informasi total agenda rapat dan informasi total pembatalan. Sedangkan konten sebelah kanan berisi daftar jadwal rapat yang sudah di-input dalam bulan berjalan. Terdapat informasi rinci mengenai tanggal rapat, jam rapat, agenda rapat, dan bagian/ bidang yang mengadakan rapat. Sedangkan Gambar-11 menunjukkan halaman *input* jadwal rapat yang mana terdapat isian nama disesuaikan dengan bidangnya, tanggal akan dilakukan rapat, waktu rapat dengan detail jam mulai dan jam selesai rapat, ruangan rapat, nama acara/agenda rapat yang akan dilaksanakan beserta keterangan atau deskripsi rapat.



Gambar-11. Halaman *input* jadwal rapat

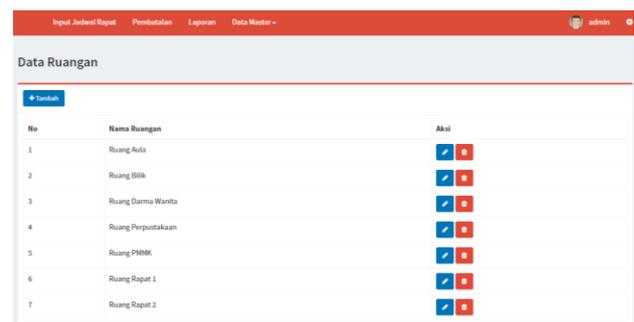
Gambar-12 menunjukkan halaman pembatalan rapat yang hanya dapat dilakukan oleh admin. Terdapat daftar rapat yang sudah di-input berisi

detail tanggal pemesanan, tanggal rapat, nama bidang, nama ruangan, waktu rapat, dan nama acara. Dalam halaman tersebut terdapat aksi untuk membatalkan rapat yakni dengan cara mengklik tombol batalkan. Sebelum jadwal rapat benar-benar terhapus, aplikasi akan memunculkan notifikasi untuk mengkonfirmasi apakah jadwal tersebut benar-benar akan dibatalkan. Apabila diklik pilihan “Ya” maka aplikasi otomatis akan menghapus data jadwal tersebut.



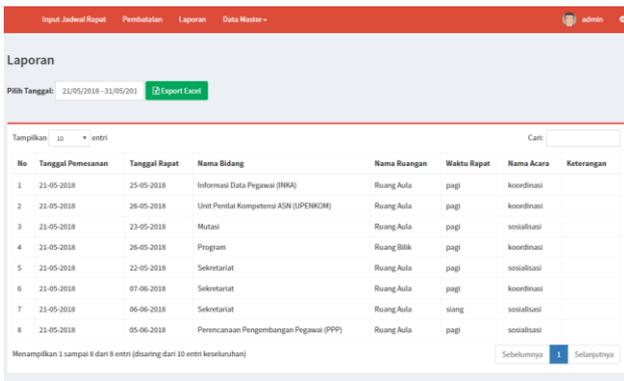
Gambar-12. Halaman pembatalan

Implementasi data master terdiri dari beberapa data yakni data ruangan, data bidang, data kepala bidang, dan data *user*. Gambar-13 menunjukkan salah satu bagian dari data master yakni data ruangan. Dalam gambar tersebut terdapat daftar ruangan yang dapat digunakan untuk rapat. Selain itu juga admin dapat menambah data ruangan dengan mengklik tombol tambah. Admin dapat mengubah dan menghapus data ruangan apabila diperlukan.



Gambar-13. Halaman data ruangan

Gambar-14 menunjukkan halaman cetak laporan yang mana terdapat pilihan tanggal atau periode cetak laporan yang diinginkan. Laporan terdiri dari beberapa bagian yakni no, tanggal pemesanan, tanggal rapat, nama bidang, nama ruangan, waktu rapat, nama acara dan keterangan. Laporan pemesanan rapat juga dapat di-export menjadi file excel untuk kemudian dapat dicetak.



Gambar-14. Halaman cetak laporan

D. Pengujian

Tahap terakhir setelah implementasi adalah tahapan pengujian. Teknik-teknik dan strategi pengujian yang dapat digunakan dipaparkan dalam [15][16]. Pengujian yang digunakan adalah *black box* untuk mengukur kinerja fungsionalitas aplikasi yang telah dibangun. Berikut adalah beberapa pengujian *black box* yang dilakukan untuk kasus uji data benar dan data salah.

Tabel-2. Pengujian *blackbox input* jadwal rapat

Kasus data dan hasil uji (data benar)	
Data masukan	Nama bidang: Sekretariat Tanggal rapat: 7 Januari 2019 Waktu rapat: Pagi (07.00-12.00) Ruang: Ruang aula Nama acara: Sosialisasi Keterangan: Wajib
Yang diharapkan	Input nama bidang, tanggal rapat, waktu rapat, ruangan, nama acara dan keterangan menampilkan hasil yang benar
Pengamatan	Hasil sesuai masukan
Kesimpulan	Pengujian diterima
Kasus data dan hasil uji (data salah)	
Data masukan	Nama bidang: Tanggal rapat: 7 Januari 2019 Waktu rapat: Pagi (07.00-12.00) Ruang: Ruang aula Nama acara: Sosialisasi Keterangan: Wajib
Yang diharapkan	Muncul pesan data nama bidang harus diisi
Pengamatan	Muncul pesan kesalahan
Kesimpulan	Pengujian diterima

Tabel-2 menunjukkan pengujian *black box* untuk *input* jadwal rapat dengan uji data benar dan data salah. Terdapat beberapa masukan seperti nama bidang, tanggal rapat, waktu rapat, ruangan, nama

acara dan keterangan. Hasil pengujian menunjukkan dengan masukan tersebut sesuai yang diharapkan dan diterima.

Tabel-3. Pengujian *black box* pembatalan jadwal rapat

Kasus data dan hasil uji (data benar)	
Data masukan	Tombol batalan dalam salah satu jadwal diklik
Yang diharapkan	Terdapat konfirmasi pembatalan jadwal, dan ketika diklik "ya" maka jadwal akan terhapus
Pengamatan	Hasil sesuai
Kesimpulan	Pengujian diterima
Kasus data dan hasil uji (data salah)	
Data masukan	Tombol batalan tidak dapat diklik
Yang diharapkan	Tidak muncul pesan konfirmasi
Pengamatan	Tidak muncul pesan konfirmasi
Kesimpulan	Pengujian diterima

Tabel-3 menunjukkan pengujian *black box* untuk pembatalan jadwal rapat, sedangkan tabel-4 menunjukkan pengujian tambah data ruangan. Kedua pengujian tersebut hasilnya sesuai yang diharapkan dan diterima. Hasil pengujian menunjukkan bahwa 100% aplikasi yang dibangun telah berjalan sesuai dengan yang diharapkan.

Tabel-4. Pengujian *black box* tambah data ruangan

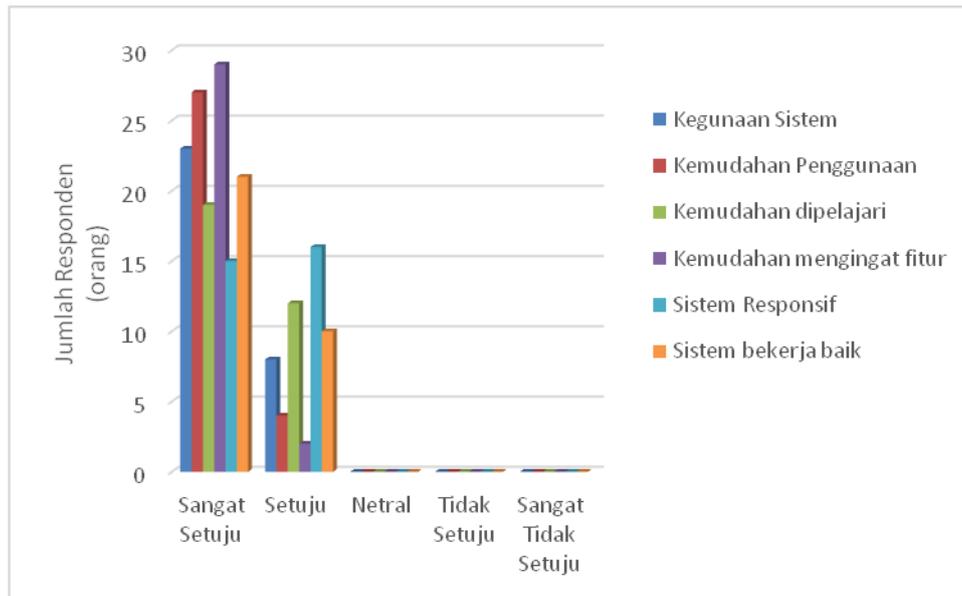
Kasus data dan hasil uji (data benar)	
Data masukan	Tombol tambah diklik dan memasukan input nama ruangan baru
Yang diharapkan	Ruangan baru ditambahkan ke basis data
Pengamatan	Hasil sesuai masukan
Kesimpulan	Pengujian diterima
Kasus data dan hasil uji (data salah)	
Data masukan	Tambah data ruangan tidak dilakukan
Yang diharapkan	Tidak terdapat penambahan data ruangan
Pengamatan	Tidak ada penambahan data
Kesimpulan	Pengujian diterima

Tahapan selanjutnya setelah semua fungsionalitas diuji dan kesimpulannya diterima. Evaluasi dilakukan untuk mengukur kinerja penggunaan dan kepuasan pengguna. Pengujian yang digunakan adalah pengujian *user acceptance testing*. Pengujian yang terkait dilakukan oleh [17]. Pengujian ini melibatkan 31 responden diantaranya adalah mahasiswa dan karyawan instansi pemerintahan. Responden diberikan 10 pertanyaan menggunakan

skala likert dengan rentang 1-5 yakni sangat setuju, setuju, netral, tidak setuju, dan sangat tidak setuju.

Evaluasi ini bertujuan mendapatkan angka kuantitatif berdasarkan persepsi pengguna dalam menggunakan aplikasi ini. Terdapat beberapa parameter yang digunakan yakni, kegunaan, kemudahan penggunaan, kemudahan dipelajari, kemudahan mengingat fitur, sistem responsif, sistem

bekerja dengan baik, kepuasan menggunakan aplikasi, antarmuka sistem menarik, fungsional sistem keseluruhan, dan pemberian rekomendasi penggunaan aplikasi. Hasil kuisioner kemudian dihitung menggunakan rumus $Y = \frac{P}{Q} \times 100\%$ dimana, Y adalah nilai presentase, P adalah jumlah responden yang menjawab pertanyaan, dan Q adalah jumlah responden.

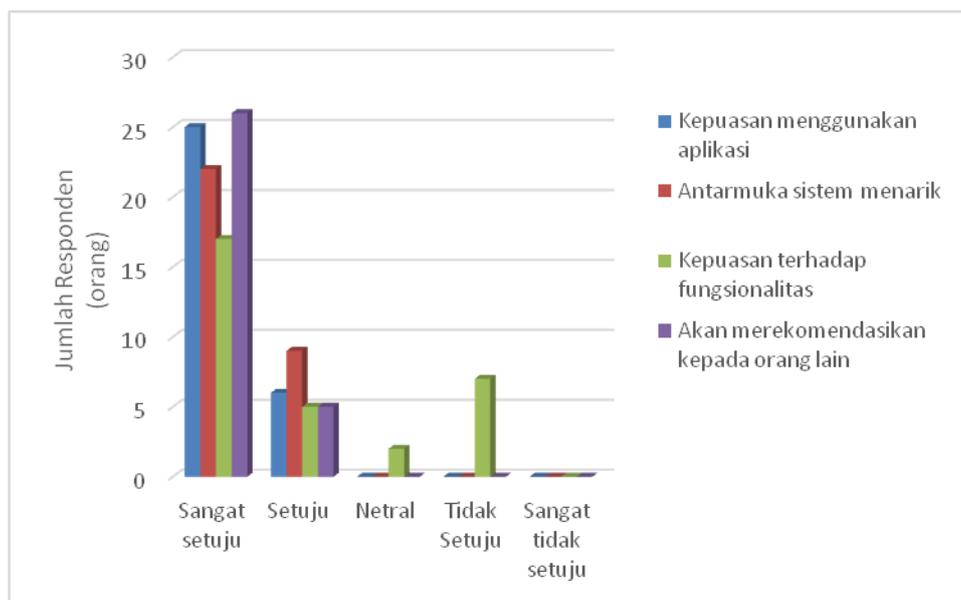


Gambar-15. Hasil evaluasi kinerja sistem

Hasil evaluasi terbagi menjadi dua bagian utama, yakni evaluasi kinerja sistem dan evaluasi kepuasan pengguna. Gambar-15 menunjukkan hasil evaluasi kinerja sistem. Terdapat 6 pertanyaan yakni kegunaan sistem, kemudahan penggunaan, kemudahan dipelajari, kemudahan mengingat fitur, sistem responsif, dan sistem bekerja baik. Grafik menunjukkan bahwa 23 responden (74%) sangat setuju dan 8 responden (26%) setuju bahwa kegunaan sistem sudah baik. Untuk kemudahan penggunaan 27 responden (87%) sangat setuju dan 4 responden (13%) setuju sistem mudah digunakan, 19 responden (61%) sangat setuju dan 12 responden (39%) setuju bahwa sistem mudah dipelajari. Kemudahan mengingat fitur sebanyak 29 responden (93,5%) sangat setuju dan 2 responden (6,5%) setuju bahwa fitur sistem mudah untuk diingat. Sistem responsif sebanyak 15 responden sangat setuju dan 16 responden setuju bahwa sistem responsif. Untuk sistem bekerja dengan baik sebanyak 21 responden (68%) sangat setuju dan 10 responden setuju sistem bekerja dengan baik. Berdasarkan penjabaran

tersebut dapat disimpulkan mayoritas responden memiliki penilaian positif terhadap kinerja sistem.

Gambar-16 menunjukkan hasil evaluasi kepuasan pengguna. Grafik menunjukkan sebanyak 25 responden (80%) sangat setuju dan 6 responden setuju bahwa mereka puas menggunakan aplikasi. Mengenai antarmuka sistem sebanyak 22 responden sangat setuju dan 9 responden setuju antarmuka sistem menarik. Untuk kepuasan terhadap fungsionalitas terdapat 71% atau sekitar 22 responden sangat setuju dan setuju bahwa mereka puas terhadap aplikasi. Akan tetapi masih terdapat 7 orang responden (22%) yang belum puas dan menginginkan adanya fitur lain yang ditambahkan seperti dapat dibuat dalam *flatfrom* android untuk permohonan jadwalnya dan notifikasi melalui sms. Sedangkan untuk pemberian rekomendasi aplikasi sebanyak 31 responden sangat setuju dan setuju bahwa mereka akan merekomendasikan aplikasi ini kepada orang lain.



Gambar-16. Hasil evaluasi kepuasan pengguna

V. PENUTUP

A. Kesimpulan

Pengembangan aplikasi penjadwalan rapat telah dilakukan dengan menggunakan metode *phased development*. Terdapat beberapa tahapan yang dilakukan meliputi perencanaan, analisis, perancangan, dan implementasi sehingga dihasilkan aplikasi versi pertama. Dalam tahap analisis dilakukan analisis kebutuhan fungsional, pemodelan dengan diagram *use case*, diagram aktivitas dan diagram kelas. Perancangan dilakukan dengan melakukan perancangan arsitektur aplikasi, perancangan basis data dan perancangan antarmuka. Implementasi dilakukan dengan menggunakan Bahasa pemrograman PHP dengan *database* MySQL. Terakhir pengujian dilakukan untuk mengukur fungsionalitas sistem versi pertama dengan menggunakan pengujian *black box*. Hasil pengujian terhadap fitur aplikasi menunjukkan bahwa aplikasi yang dibangun dapat berjalan sesuai dengan fungsi yang diharapkan. Hasil evaluasi terhadap pengguna juga telah didapatkan data bahwa mayoritas responden memiliki penilaian positif untuk kinerja sistem. Kinerja sistem dapat dikatakan 100% sudah memenuhi harapan karena berdasarkan hasil evaluasi pengguna mereka sangat setuju dan setuju terhadap semua parameter yang di evaluasi. Untuk kepuasan pengguna didapatkan 100% pengguna puas menggunakan aplikasi, antarmuka sistem menarik dan akan merekomendasikan aplikasi tersebut. Hanya saja untuk fungsionalitas sistem terdapat 7

responden yang menginginkan penambahan fitur baru.

B. Saran

Saran untuk penelitian selanjutnya adalah melakukan penambahan fitur atau fungsi lain seperti *booking* penjadwalan melalui aplikasi android, notifikasi melalui sms dan membuat antarmuka yang lebih menarik dan penambahan basis data sesuai kebutuhan.

UCAPAN TERIMA KASIH

Terimakasih kepada LPPM Universitas Dian Nuswantoro Semarang atas pemberian dana penelitian Ipteks dengan nomor kontrak 241/A.38.04/UDN-09/XI/2018 pendanaan tahun 2018, editor dan reviewer atas masukan dan sarannya.

DAFTAR PUSTAKA

- [1] Badan Pusat Statistik. Statistik Telekomunikasi Indonesia, *Telekomunikasi*. 2017.
- [2] Subhiyakto E, Kamalrudin M. Customization of Requirements Modeling Tool For Software Engineering Education, *Int. Symp. Res. Innov. Sustain*. 2014; 26 :1581–1584.
- [3] Shakshuki E.M, Hossain S.M.M. A personal meeting scheduling agent. *Personal and Ubiquitous Computing*. 2013.
- [4] Hossain S.M.M, Shakshuki E. A Negotiation Protocol for Meeting Scheduling Agent, *Procedia - Procedia Comput. Sci*. 2013; 21 (506) : 164–173.

- [5] Różycki R, Waligóra G. Power-Aware Scheduling of Preemptable Independent Jobs on a Single Processor to Meet a Schedule Deadline. *IEEE 23rd International Conference on Methods & Models in Automation & Robotics (MMAR)*. 2018: 943–946.
- [6] Shakshuki E, Koo H, Benoit D, Silver D. A distributed multi-agent meeting scheduler ☆. *Journal of Computer and System Sciences*. 2008; 74 : 279–296.
- [7] Saragih H, Hoendarto G, Reza B, Setiyadi D. Aplikasi Sistem Perangkat Lunak Menggunakan Algoritma Ant untuk Mengatur Penjadwalan Kuliah, *J. Tek. dan Ilmu Komput*. 2012; 01 (03) : 241–256.
- [8] Huaqing M. Template-based Framework for Rapid Application Development Platform. *IEEE*. 2011: 0–3.
- [9] Kosasi S., Yuliani I.D.A.E. Penerapan Rapid Application Development Pada Sistem Penjualan Sepeda Online. *J. SIMETRIS*. 2015; 6 (1) : 27–36.
- [10] Dennis A, Wixom B.H, Tegarden D. *Systems Analysis and Design with UML*, 4th Edition. John Wiley and Sons. 2012.
- [11] Abadi A, Dubinsky Y, Kirshin A, Mesika Y, I. Ben-harrush, Hadad U. *NitroGen: Rapid Development of Mobile Applications*. Proceedings conference on Systems, programming, & applications: software for humanity. 2013: 15–16.
- [12] Martínez-Fernández S., Jedlitschka A., Guzmán L., Vollmer A.M. A Quality Model for Actionable Analytics in Rapid Software Development. *IEEE, 44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, No. 732253, 2018 : 370-377.
- [13] Astuti Y.P, Subhiyakto E.R. Pengembangan Sistem Informasi Dengan Metode Waterfall Untuk Pengarsipan Data Wajib Pajak. *Techno.com*, 2017; 16 (2) : 106-113.
- [14] Subhiyakto E.R, Utomo D.W. Analisis dan perancangan aplikasi pemodelan kebutuhan perangkat lunak menggunakan metode prototyping. *Seminar Nasional Multidisiplin Ilmu Unisbank*. 2017: 2–7.
- [15] Subhiyakto E.R, Utomo D.W. Software Testing Techniques and Strategies Use in Novice Software Teams. *Jurnal SISFO : Inspirasi Profesional Sistem Informasi*. 2016; 5 (5) : 556-562.
- [16] Subhiyakto E.R, Utomo D.W. Strategi, teknik, faktor pendukung dan penghambat pengujian untuk pengembang perangkat lunak pemula. *Seminar Nasional Teknologi Informasi dan Komunikasi*. No. Sentika, 2016; 236–241.
- [17] Iskandar D, Nofiyanti. Performa Testing unuk Mengetahui Reabilitas QoS (Quality of Service) Website Fakultas Teknik Universitas Jenderal Soedirman. *Dinamika Rekayasa*. 2018; 14 (1) : 39–46.

