

Pengembangan Sistem Manajemen Bimbingan di Program Studi Teknik Informatika Universitas Pamulang

Suci Fitriyana¹, and Aries Saifudin²

^{1,2}Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Banten, Indonesia
e-mail: ¹sucifitriyana99@gmail.com, ²aries.saifudin@gmail.com

Abstract

Time becomes the main problem in doing guidance. Because students and lecturers collide with each other in time to conduct guidance, which makes the completion of the student's final assignment delayed. So that it makes it difficult for supervisors and study program officials to see the development of students who do guidance. Application to deal with time clashing with each other in guidance and the difficulty of seeing the development of guidance students. Thus, a guidance management system is created that can reduce the problem. This management system makes it easy for students and lecturers to carry out guidance at any time without having to clash time and come to campus, supervisors and study program officials are easy to see the development of student guidance without having to meet directly.

Keywords: Student, Lecturer, Guidance, Time

1. Pendahuluan

Manajemen bimbingan adalah segala usaha atau cara untuk mengelola bimbingan agar mencapai hasil yang optimal dan untuk menentukan hasil yang berkualitas, terkontrol yang sesuai dengan prosedur. Mengelola pelayanan bimbingan agar dapat mencapai tujuan yang telah ditetapkan (Zamroni, 2015). Dalam manajemen bimbingan sudah membuat perencanaan yang harus dilaksanakan, dievaluasi, dianalisis, dan ditindaklanjuti yang direncanakan mulai dari pelayanan dasar bimbingan (Ratnawulan, 2016). Manajemen Bimbingan sangat penting karena untuk menentukan hasil penulisan yang berkualitas dan mencapai standar penulisan yang baik dan benar.

Manajemen bimbingan saat ini masih menggunakan sistem yang konvensional yakni dengan mendatangi kampus untuk menemui dosen, guna melakukan bimbingan dengan membawa skripsi atau laporan kerja praktek yang sudah dicetak. Bimbingan dilakukan hingga skripsi atau laporan kerja praktek mahasiswa dianggap layak oleh dosen pembimbing.

Dengan sistem yang masih konvensional membuat mahasiswa dan dosen kesulitan dalam melakukan bimbingan. Yaitu sulitnya menyamakan waktu antara mahasiswa dan dosen. Karena mahasiswa yang bekerja dan dosen yang mempunyai kesibukan kerja di tempat lain. Dengan sistem yang masih konvensional membuat pihak struktural program studi tidak bisa mengontrol

progress bimbingan mahasiswa yang lambat dalam mengerjakan skripsi atau laporan kerja praktek.

Oleh karena itu sistem manajemen bimbingan dapat memudahkan antara mahasiswa dengan dosen untuk bisa sering berinteraksi dalam bimbingan tanpa harus menyamakan waktu. Dosen dapat secara rutin dan mahasiswa hingga bisa terlihat progressnya. Mahasiswa bisa melakukan bimbingan skripsi dan laporan kerja praktek kapan saja. Dosen dapat melihat progress mahasiswa secara rutin tanpa harus bertemu langsung. Mahasiswa bisa melakukan bimbingan skripsi kapan saja. Dosen dapat membimbing mahasiswa melalui media online kapan saja sesuai dengan waktu yang tersedia. Pejabat program studi bisa melihat progress bimbingan dengan rutin.

Metode yang digunakan ialah dengan dibuatnya sistem manajemen bimbingan dapat memudahkan mahasiswa melakukan bimbingan skripsi maupun laporan kerja praktek, tanpa harus mendatangi kampus. Dan dosen bisa melihat perkembangan mahasiswa melalui sistem manajemen bimbingan tanpa harus bertatap muka langsung. Dimulai dari pengajuan judul sampai pengumuman sidang bisa dilihat melalui *website*. Metode ini pernah digunakan oleh Utriani untuk melihat perkembangan bimbingan skripsi. Bahwa dengan dibuatnya aplikasi melihat perkembangan bimbingan skripsi ini memudahkan mahasiswa untuk melakukan bimbingan tanpa harus bertemu dosen (Utriani, 2017). Dan pernah digunakan juga oleh Azrani Sari dalam penelitian di Universitas Mulawarman. Bahwa sistem informasi bimbingan

ini sangat membantu dalam memberikan layanan bimbingan tugas akhir, sehingga dalam mengerjakan dan menyelesaikan tugas akhir dapat dilaksanakan dengan lebih baik dibandingkan dengan sistem sebelumnya (Sari, 2017). Namun kekurangan dari sistem bimbingan menggunakan website ini membuat mahasiswa tidak sering mendatangi kampus. Tidak bisa bertanya secara jelas dengan dosen pembimbing tersebut. Dan jarang menemui dosen pembimbing.

2. Metodologi

Analisa sistem saat ini dapat diartikan sebagai kegiatan penguraian suatu sistem yang sedang berjalan secara utuh ke dalam bagian komponen-komponennya dengan tujuan untuk mengidentifikasi dan mengevaluasi hambatan, masalah, dan kesempatan yang ada serta kebutuhan yang diharapkan. Sehingga berdasarkan hasil analisa sistem dapat diusulkan perancangan aplikasi yang dibutuhkan. Aktivitas analisa sistem dilakukan setelah perencanaan (*planning*) dan sebelum pengembangan sistem/aplikasi. Analisa sistem berfungsi untuk mengetahui bagaimana suatu sistem itu bekerja. Aktivitas analisa sistem merupakan tahap pengembangan sistem yang sangat penting dan kritis, karena jika ada kesalahan di tahap ini dapat menyebabkan kesalahan pada tahap selanjutnya. Sebab hasil dari analisa akan dijadikan sebagai bahan uji dan analisa pada pengembangan dan penerapan sistem/aplikasi yang diusulkan. Gambaran sistem saat ini ditunjukkan pada Gambar 1.

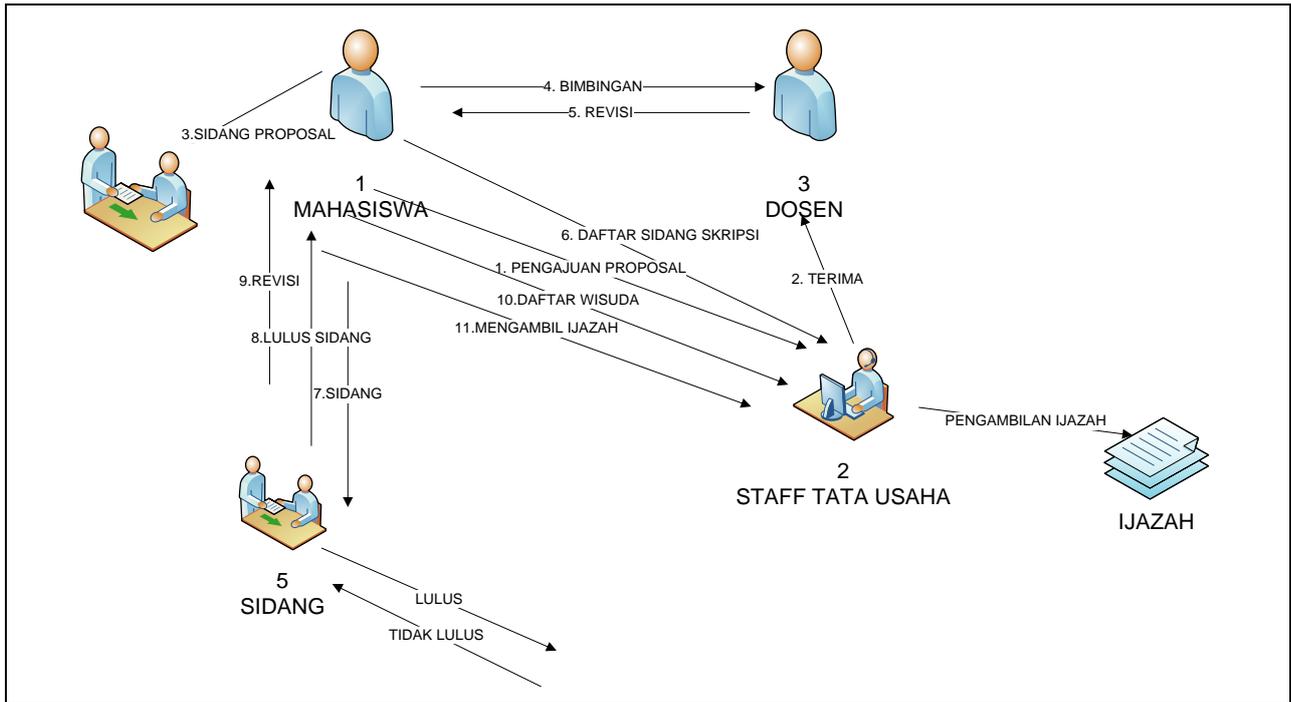
Sistem yang diusulkan merupakan aplikasi web yang akan dikembangkan menggunakan framework *Code Igniter*. *Code Igniter* merupakan salah satu *Framework* untuk PHP dan telah dianggap yang paling baik pada saat ini, karena memiliki fitur yang lengkap untuk pengembangan aplikasi berbasis *web*. Selain itu, saat ini banyak pengembang aplikasi web yang menggunakan *framework Code Igniter*, sehingga mudah untuk mencari referensi dan forum diskusi.

Perancangan aplikasi didasarkan pada aplikasi/sistem yang akan dibuat. Karena aplikasi yang diusulkan menggunakan basis data, maka pertama akan dibuat rancangan basis datanya terlebih dulu. Setelah selesai rancangan basis data akan dilanjutkan dengan membuat rancangan aplikasi.

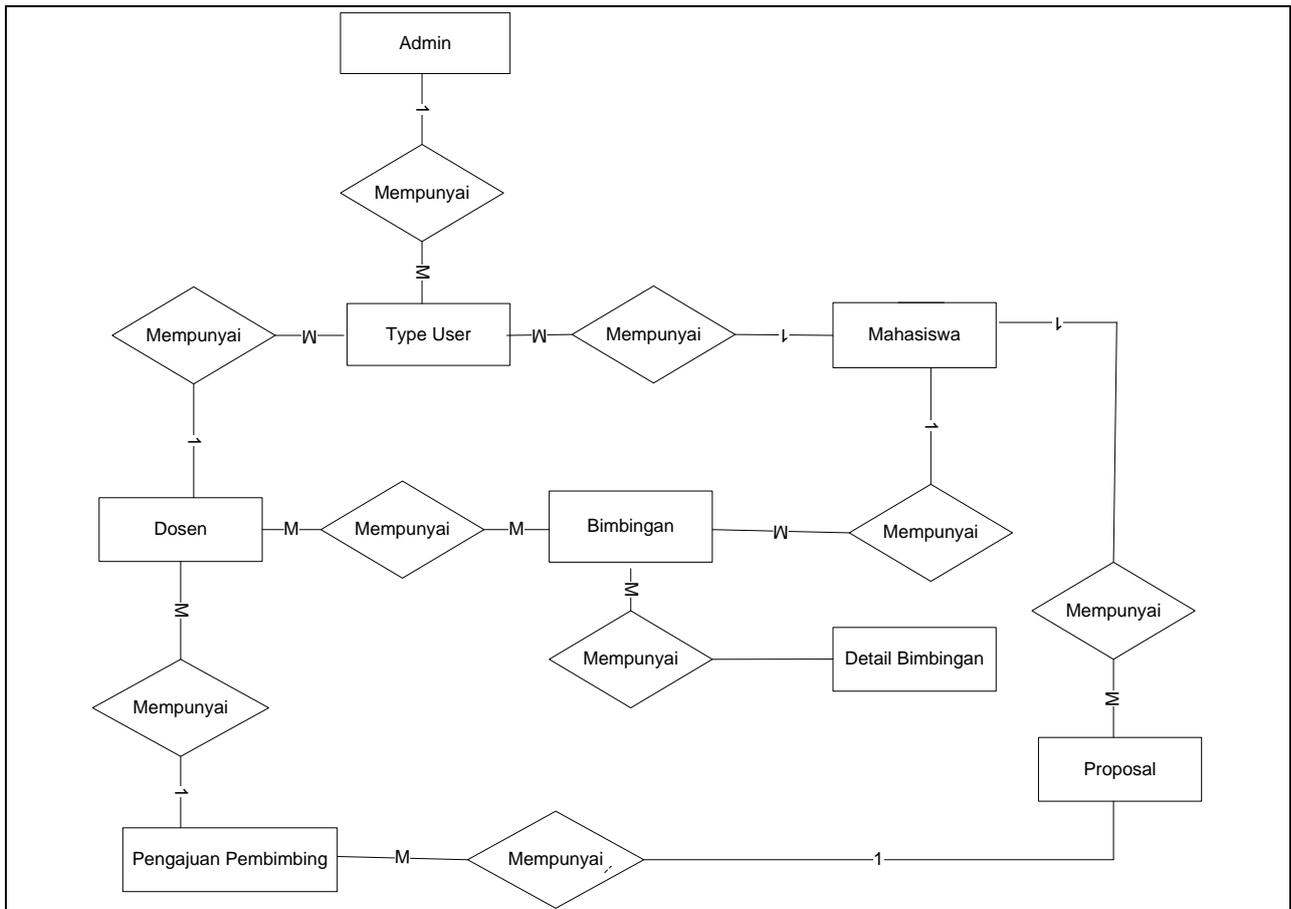
Pada umumnya perancangan basis data dilakukan dengan tujuan untuk memberikan gambaran secara umum mengenai basis data yang akan digunakan dalam aplikasi/sistem yang dibuat. Rancangan basis data ini digunakan untuk mengidentifikasi entitas-entitas dan relasinya dalam sistem yang akan dibuat. Setelah tahap analisa terhadap sistem yang sedang berjalan saat ini selesai dilakukan akan dilanjutkan dengan membuat rancangan basis data dan aplikasi. Perancangan aplikasi merupakan aktivitas yang menerapkan berbagai teknik dan prinsip untuk tujuan mendefinisikan suatu perangkat lunak atau sistem secara detail dan memadai sehingga memungkinkan untuk direalisasikan fisiknya.

Rancangan basis data dapat digambarkan menggunakan ERD (*Entity Relational Diagram*) maupun LRS (*Logical Record Structure*). ERD atau diagram-ER merupakan diagram untuk memodelkan atau menggambarkan hubungan antarentitas yang digunakan dalam aplikasi. Rancangan basis data aplikasi yang akan dibuat digambarkan menggunakan ERD seperti ditunjukkan pada Gambar 2.

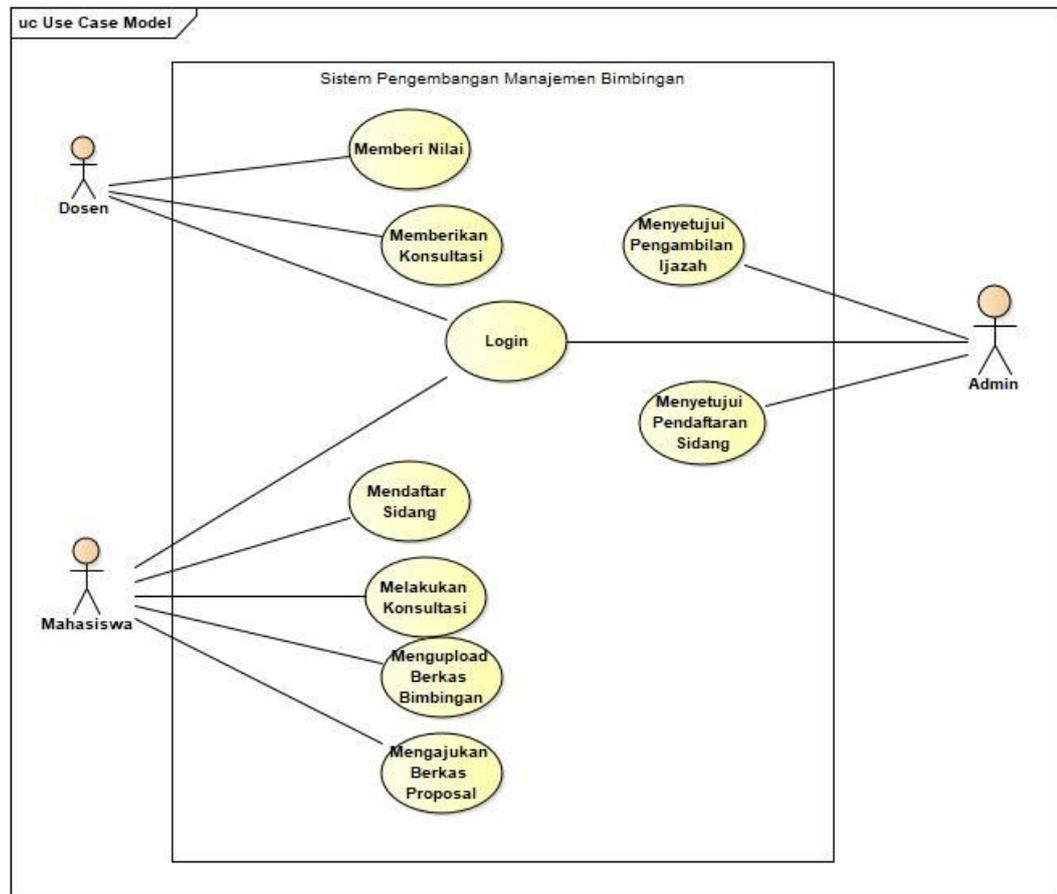
Perancangan sistem merupakan aktivitas untuk membuat rancangan atau desain suatu sistem/aplikasi yang mencakup langkah-langkah operasi sistem dalam proses pengolahan data, hubungan antarkomponen dalam sistem, serta prosedur untuk mendukung pengoperasian sistem. Pada perancangan sistem/aplikasi yang diusulkan digunakan UML (*Unified Modeling Language*). UML merupakan teknik pemodelan yang mengolaborasi antara metode-metode Booch, OOSE (*Object Oriented Software Engineering*), OMT (*Object Modeling Technique*) dan beberapa metode lainnya.



Gambar 2 Sistem yang Berjalan Saat Ini



Gambar 3 Entity Relational Diagram (ERD) Sistem Usulan



Gambar 3 Use Case Diagram Sistem Manajemen Bimbingan yang Diusulkan

Saat ini UML merupakan pemodelan yang paling banyak digunakan dalam menganalisa dan merancang sistem karena sesuai dengan konsep pemrograman berorientasi objek. UML banyak mengadaptasi penggunaan bahasa pemrograman berorientasi objek (OOP/*Object Oriented Programming*). UML Versi 2 terdiri dari 13 jenis diagram resmi (Fowler, 2003) walaupun dalam perancangan tidak harus dibuat semuanya. Beberapa literatur menyatakan bahwa beberapa diagram UML dapat digabung sehingga menjadi delapan diagram, misalnya *communication diagram*, *sequence diagram* dan *timing diagram* digabung menjadi *interaction diagram* (Nugroho, 2009).

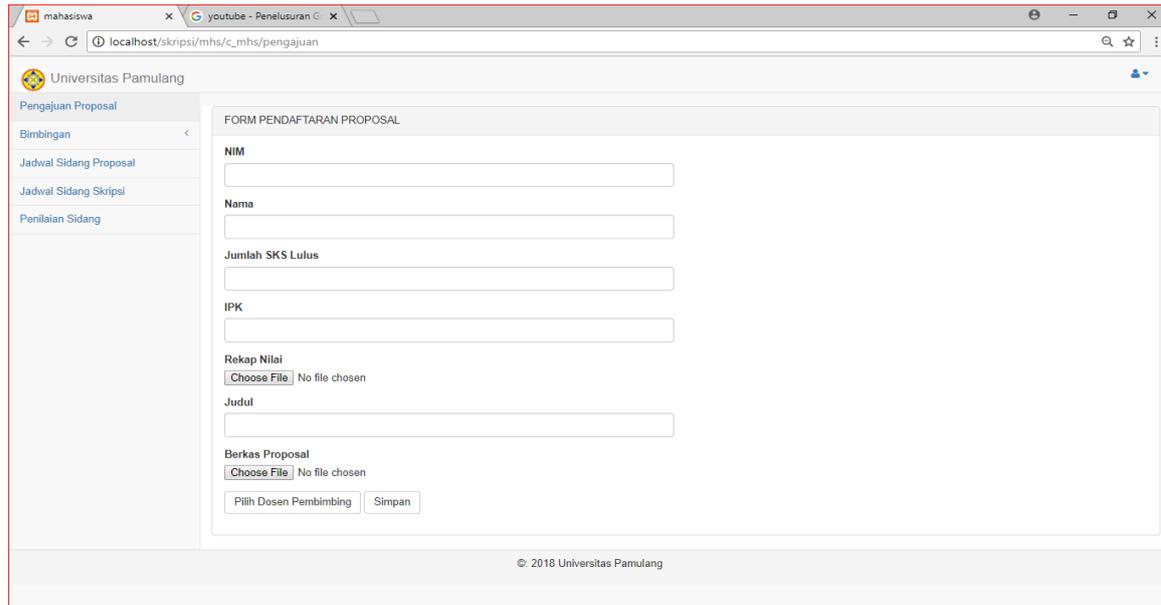
Use Case Diagram merupakan gambaran skenario dan interaksi antara pengguna dengan sistem. Sebuah *use case diagram* menggambarkan hubungan antara aktor (pengguna) dan kegiatan yang dapat dilakukannya terhadap aplikasi. *Use case diagram* sistem usulan untuk memperlihatkan peranan aktor dalam interaksinya dengan sistem sistem usulan ditunjukkan pada Gambar 3.

3. Implementasi dan Pengujian Implementasi

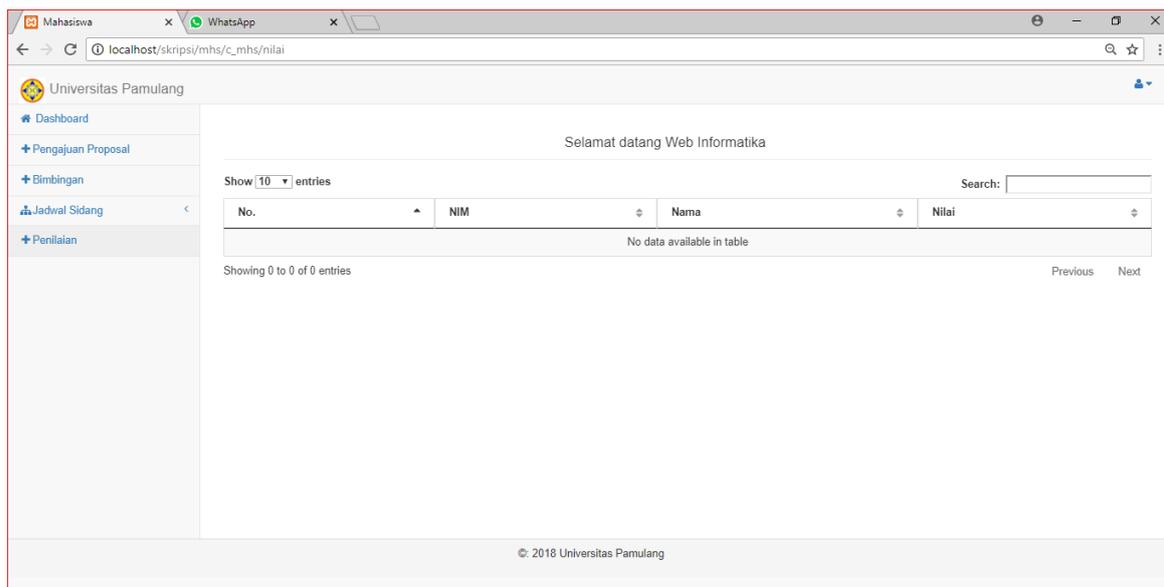
Setelah tahap perancangan selesai dilanjutkan dengan tahap implementasi. Implementasi sistem merupakan tahap pembuatan perangkat lunak berdasarkan hasil perancangan sistem. Tahap implementasi merupakan aktivitas menerjemahkan hasil perancangan sistem ke dalam bahasa pemrograman yang digunakan dan dapat dimengerti oleh mesin.

Pada penelitian ini, sistem yang diusulkan dikembangkan menggunakan *Framework Code Igneter Versi 3.1.3*. Untuk penyusunan kode program digunakan aplikasi *Sublime Text 3*. Aplikasi untuk server digunakan paket XAMPP yang telah mencakup MySQL sebagai database dan Apache sebagai servernya.

Setelah mengimplementasikan desain basis data dan aplikasi, antarmuka aplikasi yang dibuat memiliki tampilan yang ditunjukkan pada Gambar 4 dan Gambar 5. Gambar 4 menunjukkan tampilan pengajuan proposal untuk mahasiswa. Mahasiswa dapat mengisi sesuai dengan form yang ada dengan lengkap. Sedangkan Gambar 5 menunjukkan tampilan untuk melihat nilai tugas akhir.



Gambar 4 Pengajuan Proposal Mahasiswa



Gambar 5 Penilaian Mahasiswa

Pengujian perangkat lunak merupakan tahapan terpenting dalam pengembangan perangkat lunak yang bertujuan untuk menemukan cacat atau kesalahan pada sistem (perangkat lunak) yang diuji. Pengujian digunakan untuk mencari kesalahan/cacat dan memperbaikinya, sehingga dapat meningkatkan kualitas software/aplikasi yang dibuat (Aries & Wahono, 2015). Pengujian merupakan tahapan pengembangan perangkat lunak yang bersifat integral dan dapat menghabiskan setengah dari waktu pengembangan perangkat lunak (Simarmata, 2010). Pengujian

merupakan proses yang banyak memakan waktu pengembangan perangkat lunak karena dapat menghabiskan sampai 50% dari jadwal proyek sehingga menjadi tahapan pengembangan sistem paling mahal (Fakhrahmad & Sami, 2009).

Adapun teknik pengujian yang digunakan untuk memverifikasi perangkat lunak yang manajemen bimbingan adalah pengujian *black box*. Teknik pengujian *black box* merupakan teknik untuk memverifikasi perangkat lunak yang berfokus pada spesifikasi fungsional dari perangkat lunak.

Pengujian *Black Box*

Pengujian *black box* difokuskan untuk memverifikasi fungsi dari perangkat lunak. Pada pengujian *black box*, penguji dapat mendefinisikan kumpulan masukan, keluaran dan kondisi perangkat lunak yang diharapkan untuk menguji spesifikasi fungsional perangkat lunak. Pengujian *black box* biasa digunakan untuk memverifikasi tentang hal-hal berikut:

- a. Fungsional perangkat lunak yang belum ada, atau ada kesalahan.
- b. Kesalahan pada antarmuka.
- c. Adanya kesalahan struktur data atau akses ke basis data.
- d. Kinerja perangkat lunak yang tidak sesuai kebutuhan.
- e. Terminasi perangkat lunak dan kesalahan inisialisasi.

- a. Pengujian *black box* pada halaman login

Pengujian *black box* bertujuan untuk menguji perangkat lunak yang dikembangkan apakah sudah sesuai dengan spesifikasi fungsional sistem yang telah didefinisikan. Pengujian *black box* juga difungsikan untuk memverifikasi fungsi-fungsi khusus yang ada pada perangkat lunak yang dikembangkan. Pada pengujian *black box*, kebenaran suatu perangkat lunak yang diuji hanya didasarkan pada keluaran yang dihasilkan dan kondisi perangkat lunak ketika diberikan masukan untuk fungsi yang ada tanpa memperhatikan bagaimana alur pemrosesan untuk menghasilkan keluaran tersebut.

Untuk menguji aplikasi manajemen bimbingan yang telah dibuat diterapkan teknik pengujian *black box*. Beberapa hasil pengujian ditunjukkan pada Tabel 1 sampai Tabel 3 sebagai berikut:

Tabel 1 Pengujian Black Box Halaman Login Testing

No	Deskripsi Pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Mengosongkan <i>username</i> dan langsung menekan tombol login	Sistem akan menolak akses login dan akan menampilkan “ <i>please fill out this field</i> ”	Sistem dapat menolak akses login dan akan menampilkan “ <i>please fill out this field</i> ”	Valid
2	Mengosongkan <i>password</i> dan langsung menekan tombol login	Sistem akan menolak akses login dan akan menampilkan “ <i>please fill out this field</i> ”	Sistem dapat menolak akses login dan akan menampilkan “ <i>please fill out this field</i> ”	Valid
3	Mengisi <i>username</i> yang salah dan klik tombol login	Sistem akan menolak akses login dan akan menampilkan “user tidak ditemukan”	Sistem dapat menolak akses login dan akan menampilkan “user tidak ditemukan”	Valid
4	Mengisi <i>username</i> dan <i>password</i> yang benar lalu menekan tombol login	Sistem akan menampilkan halaman	Sistem dapat menampilkan halaman	Valid

- b. Pengujian *black box* pada master data mahasiswa

Tabel 2 Pengujian Black Box pada Master Data Mahasiswa

No	Skenario pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	<i>User</i> mengisi <i>form</i> data mahasiswa dengan benar dan menekan tombol simpan	Sistem akan menampilkan data yang dimasukkan ke dalam tabel	Sistem dapat menampilkan data di dalam tabel	Valid
2	<i>User</i> mengosongkan <i>form</i> data mahasiswa dan menekan tombol simpan	Sistem akan menolak dan memberikan keterangan “ <i>please fill out this field</i> ”	Sistem dapat menolak dan memberikan keterangan “ <i>please fill out this field</i> ”	Valid
3	<i>User</i> mengedit data pada table master data mahasiswa	Sistem berhasil mengedit, maka dalam tabel nama berubah	Sistem dapat mengedit, maka dalam tabel nama berubah	Valid
4	<i>User</i> menghapus data mahasiswa pada tabel	Sistem akan memberikan keterangan “anda yakin hapus data ini”	Sistem dapat memberikan keterangan “anda yakin hapus data ini”	Valid

c. Pengujian *black box* pada master data dosen

Tabel 3 Pengujian Black Box pada Master Data Dosen

NO	Skenario pengujian	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	User mengisi form data dosen dengan benar dan menekan tombol simpan	Sistem akan menampilkan data yang dimasukkan di dalam tabel	Sistem dapat menampilkan data yang dimasukkan di dalam tabel	Valid
2	User mengosongkan form data dosen dan menekan tombol simpan	Sistem akan menolak dan memberikan keterangan "please fill out this field"	Sistem dapat menolak dan memberikan keterangan "please fill out this field"	Valid
3	User mengedit data pada table master data dosen	Sistem bisa mengedit, maka dalam tabel nama berubah	Sistem berhasil mengedit, maka dalam tabel nama berubah	Valid
4	User menghapus data dosen pada tabel	Sistem akan memberikan keterangan "anda yakin hapus data ini"	Sistem dapat memberikan keterangan "anda yakin hapus data ini"	Valid

Berdasarkan hasil pengujian aplikasi manajemen bimbingan menggunakan teknik pengujian *black box* menunjukkan bahwa tidak ditemukan kesalahan/cacat. Secara fungsional aplikasi manajemen bimbingan yang dikembangkan telah memenuhi persyaratan yang ditetapkan. Aplikasi manajemen bimbingan yang dikembangkan dapat digunakan untuk mengelola data mahasiswa, dosen dan administrasi bimbingan skripsi.

4. Kesimpulan

Dengan sistem ini mahasiswa dapat dengan mudah melakukan bimbingan dengan dosen pembimbing di mana pun dan kapan pun tanpa harus terbatas oleh waktu dan tempat. Dengan sistem ini dosen dapat dengan mudah melihat perkembangan mahasiswa tanpa harus bertatap muka langsung dengan mahasiswanya dan dapat melihat perkembangan mahasiswa yang dibimbingnya di mana pun dan kapan pun.

Referensi

- Aries, S., & Wahono, R. S. (2015). Pendekatan Level Data untuk Menangani Ketidakseimbangan Kelas pada Prediksi Cacat Software. *Journal of Software Engineering*, 1(2), 76-85.
- Fakhrhmad, S. M., & Sami, A. (2009). Effective Estimation of Modules' Metrics in Software Defect Prediction. *Proceedings of the World Congress on Engineering* (pp. 206-211). London: Newswood Limited.
- Fowler, M. (2003). *UML Distilled: A Brief Guide to the Standard Object Modeling Language (Third Edition)*. Boston: Addison-Wesley Professional.
- Nugroho, A. (2009). *Rekayasa Perangkat Lunak Menggunakan UML dan Java*. Yogyakarta: Yogyakarta.
- Ratnawulan, T. (2016). Manajemen Bimbingan Konseling SMP Kota dan Kabupaten Bandung. *Universitas Islam Nusantara Bandung*, 1 - 17.
- Sari, A. (2017). Teknik Informatika, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Mulawarman. *Sistem informasi Bimbingan Tugas Akhir Pada Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Mulawarman*, 243-249.
- Simarmata, J. (2010). *Rekayasa Perangkat Lunak*. Yogyakarta: Andi.
- Utriani. (2017). Monitoring Bimbingan Skripsi Online. *Sekolah Tinggi Ilmu Komputer Palang Karaya*, 34 - 47.
- Zamroni, E. (2015). Manajemen Bimbingan dan Konseling Berbasis Permendikbud. *FKIP Universitas Muria Kudus*, 1-11.