

Perancangan Sistem Informasi Sebaran Dosen Pembimbing Skripsi dan Tesis Berbasis Website di Rumpun Matematika FMIPA UNJ

Trisna Hastuti Puspita Ningrum¹, Fariani Hermin Indiyah², Ari Hendarno³
Program Studi Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Negeri Jakarta, Jakarta Timur, Indonesia
trisnahastutipuspita@gmail.com¹, fhhermin@unj.ac.id², arihendarno@unj.ac.id³

Abstrak— Penyelesaian studi mahasiswa pada Rumpun Matematika FMIPA UNJ melalui jalur skripsi untuk jenjang sarjana dan tesis untuk jenjang magister. Proses penyusunan tugas akhir dibimbing oleh 2 (dua) orang dosen yang ditentukan oleh Koordinator Program Studi. Kriteria penentuan dosen pembimbing adalah (1) kesesuaian bidang keahlian, (2) payung penelitian dosen, dan (3) beban bimbingan tugas akhir dosen. Ketiga kriteria tersebut haruslah menjadi acuan Koordinator Program Studi saat menentukan sebaran dosen pembimbing. Skripsi ini bertujuan untuk membangun suatu sistem informasi sebaran dosen pembimbing berbasis *website* yang dapat membantu Koordinator Program Studi menentukan sebaran dosen pembimbing ketika mencocokkan judul tugas akhir mahasiswa dengan kriteria penentuan dosen pembimbing. Sistem informasi ini dikembangkan menggunakan metode pengembangan perangkat lunak SDLC dengan Model *Waterfall*, *framework* Laravel 6 untuk *back-end* dan *framework* Bootstrap untuk *front-end*. Pada akhir pengembangan dilakukan pengujian menggunakan UAT (*User Acceptance Test*) baik secara fungsional maupun kebergunaannya, dapat dikatakan bahwa Perancangan Sistem Informasi Sebaran Dosen Pembimbing Skripsi dan Tesis telah berjalan dengan baik dan sesuai yang diharapkan dengan tingkat kebergunaan sebesar 85,4% pada keseluruhan sistem.

Kata kunci—Sistem Informasi, *website*, tugas akhir, SDLC, Model *Waterfall*, UAT.

I. PENDAHULUAN

Penentuan dosen pembimbing tugas akhir mahasiswa dilakukan oleh Koorprodi dengan memperhatikan bidang keahlian dosen, banyak mahasiswa bimbingan, dan payung penelitian dosen.

Penulis bertanya kepada 4 Koorprodi pada Rumpun Matematika FMIPA UNJ mengenai penentuan dosen pembimbing tersebut dan 3 Koorprodi menjawab terdapat permasalahan pada proses sebaran dosen pembimbing diantaranya adalah (1) beban bimbingan tugas akhir dosen yang tidak merata, dikarenakan masih manualnya perekaman data dalam file-file terpisah, sehingga menyulitkan saat proses rekapitulasi beban bimbingan, (2) data rekapitulasi beban bimbingan dosen tidak tersimpan secara daring menyebabkan Koorprodi kesulitan untuk melihat rekapitulasi beban bimbingan dosen secara *realtime*, (3) dan diperlukan waktu yang cukup lama untuk menentukan dosen pembimbing tugas akhir.

Selain itu, pada saat proses bimbingan tugas akhir berlangsung antara dosen dan mahasiswa bimbingannya, pencatatan kemajuan bimbingan efisien dilakukan dengan *form* manual, untuk jenjang sarjana menggunakan kartu bimbingan sedangkan jenjang magister menggunakan buku konsultasi. Penulis bertanya kepada 10 mahasiswa yang sedang mengerjakan tugas akhir mengenai pencatatan kemajuan bimbingan dengan *form* manual tersebut, dan para mahasiswa menjawab terdapat kendala diantaranya yaitu (1) dapat tertinggal ketika bimbingan dan hilang karena berbentuk media cetak, (2) dapat rusak dan kotor jika terkena cairan, (3) kolom-kolom pada *form* manual terlalu kecil untuk menuliskan keterangan bimbingan, (4) dan tidak bisa menggunggah berkas proposal untuk dosen memberi tanda dimana saja yang perlu diperbaiki.

Penelitian terkait permasalahan kartu bimbingan telah dilakukan oleh Tiara Amelia (2017) dalam skripsi yang berjudul “*Monitoring* Skripsi Mahasiswa Berdasarkan Kartu Bimbingan Skripsi Berbasis Web”. Akan tetapi penelitian tersebut terbatas pada *monitoring* skripsi secara daring.

Salah satu upaya yang dapat membantu dalam proses penyebaran dosen pembimbing tugas akhir yaitu

dengan merancang suatu sistem informasi sebaran dosen pembimbing skripsi dan tesis yang dapat diakses secara daring, dan penelitian ini juga sebagai lanjutan dari penelitian Tiara Amelia (2017).

II. KAJIAN PUSTAKA

A. Sistem Informasi

Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengelolaan transaksi harian, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang dibutuhkan.[1]

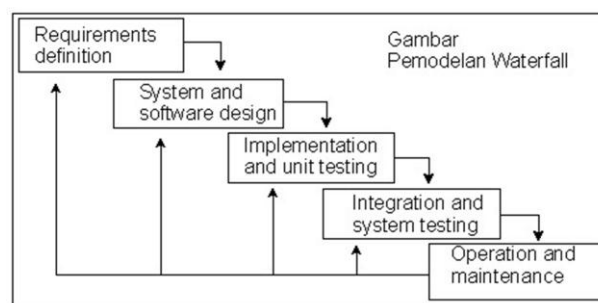
B. Sistem Informai Sebaran Dosen Pembimbing Skripsi dan Tesis

Sistem informasi sebaran dosen pembimbing skripsi dan tesis adalah suatu sistem yang bertujuan untuk mengatasi permasalahan sebaran dosen pembimbing, *monitoring* tugas akhir dan melakukan pencatatan kemajuan bimbingan mahasiswa. Data yang terdapat pada sistem informasi sebaran dosen pembimbing skripsi dan tesis dapat diperbarui (*update*) oleh *user* (Koordinator Program Studi, Dosen Mahasiswa dan Admin) sendiri. Sistem informasi sebaran dosen pembimbing skripsi dan tesis ini mencakup data diri, data mahasiswa bimbingan, dan profil dosen dari setiap Program Studi.

C. Software Development Life Cycle

Menurut Mohit Kumar Sharma (2017) SDLC adalah suatu konsep penting yang digunakan dalam rekayasa perangkat lunak yang menjelaskan prosedur perencanaan, pembuatan, pengkodean, pengujian, dan implementasi dari spesifikasi kebutuhan pengguna.[2]

Terdapat berbagai model pada SDLC seperti *Waterfall Model*, *Rapid Application Development (RAD) Model*, *V Model*, *Iterative Model*, *Agile Model*, *Extreme Programming*, *Scrum*, *Spiral Model*, dan lain sebagainya. Pada penelitian ini model yang akan digunakan dalam pengembangan perangkat lunak yaitu *Waterfall Model*. Dipilihnya Model *Waterfall* karena tahapan pada model ini berurutan dan berkelanjutan dimana tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya. Berikut tahapan yang terdapat pada Model *Waterfall*: [3]



Model Waterfall

1. Requirements Definition

Proses pencarian kebutuhan diintensifkan dan difokuskan pada *software*. Untuk mengetahui sifat dari program yang akan dibuat, maka para *software engineer* harus mengerti tentang domain informasi dari *software*, misalnya fungsi yang dibutuhkan yaitu *user interface*. Dari aktivitas tersebut harus didokumentasikan dan ditunjukkan kepada *client*.

2. System and Software Design

Proses ini digunakan untuk mengubah kebutuhan-kebutuhan diatas menjadi representasi ke dalam bentuk "*blueprint*" *software* sebelum pengkodean dimulai. Desain harus dapat mengimplementasikan kebutuhan yang telah disebutkan pada tahap sebelumnya. Seperti aktivitas sebelumnya, maka proses ini juga harus didokumentasikan sebagai konfigurasi dari *software*.

3. Implementation and unit Testing

Untuk dapat dimengerti oleh mesin, dalam hal ini adalah komputer, maka desain yang telah dibuat harus diubah menjadi bentuk yang dapat dimengerti oleh mesin, yaitu ke dalam bahasa pemrograman melalui proses pengkodean. Tahap ini merupakan implementasi dari tahap desain yang secara teknis nantinya dikerjakan oleh *programmer*.

4. Integration and system testing

Sesuatu yang dibuat haruslah diuji obakan. Demikian juga dengan *software*. Semua fungsi-fungsi *software* harus diujicobakan, agar *software* bebas dari *error*, dan hasilnya harus benar-benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya.

5. *Operation and maintenance*

Pemeliharaan suatu *software* diperlukan, termasuk di dalamnya adalah pengembangan, karena *software* yang dibuat tidak selamanya hanya seperti itu. Ketika dijalankan mungkin masih terdapat *error* kecil yang tidak ditemukan sebelumnya, atau ada penambahan fitur-fitur yang belum ada pada *software* tersebut. Pengembangan diperlukan ketika adanya perubahan dari eksternal perusahaan seperti ketika ada pergantian sistem operasi, atau perangkat lainnya.

D. *Unified Modelling Language (UML)*

UML adalah suatu bahasa yang banyak digunakan untuk menentukan, memvisualisasikan, membangun dan mendokumentasikan suatu sistem informasi.[4] Beberapa jenis UML yang digunakan dalam penelitian ini sebagai berikut:

1. *Use Case Diagram*

Diagram model *use case* merepresentasikan satu tujuan dari sistem dan menggambarkan suatu rangkaian kegiatan serta interaksi pengguna untuk mencapai tujuan.[4]

2. *Activity Diagram*

Activity Diagram atau Diagram Aktifitas secara grafis digunakan untuk menggambarkan rangkaian aliran aktifitas baik proses bisnis ataupun *use case*. [4]

3. *Class Diagram*

Class diagram menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan. [5]

4. *Sequence Diagram*

Sequence diagram atau diagram urutan adalah salah satu jenis diagram yang paling populer dari tiga tipe diagram interaksi, karena dapat menunjukkan jenis informasi secara sederhana dan tepat. [6]

E. *Database*

Menurut Ni Ketut Dewi Ari Jayanti dan Ni Kadek Sumiarti basis data dapat didefinisikan sebagai sekumpulan data yang terintegrasi, yang diorganisasi untuk memenuhi kebutuhan para pemakai di dalam suatu organisasi. Makna dari terintegrasi yaitu setiap data akan memiliki hubungan dengan data yang lainnya.

Dalam merancang *database*, yang perlu dilakukan yaitu membuat ERD. ERD (*Entity Relationship Diagram*) merupakan notasi grafis dalam pemodelan data konseptual yang mendeskripsikan hubungan antar penyimpanan. ERD digunakan untuk memodelkan struktur data dan hubungan antar data.

F. *Framework*

Framework adalah kerangka kerja. *Framework* juga dapat diartikan sebagai kumpulan *script* (terutama *class* dan *function*) yang dapat membantu *developer* dalam menangani berbagai masalah dalam pemrograman seperti koneksi ke *database*, pemanggilan *variabel*, dan *file*. Sehingga *developer* lebih fokus dan lebih cepat membangun aplikasi. Secara sederhana dapat dikatakan bahwa *framework* adalah kumpulan fungsi (*libraries*) sehingga seorang *programmer* tidak perlu lagi membuat fungsi-fungsi dari awal dan biasanya disebut kumpulan *library*. [7]

III. IMPLEMENTASI PROGRAM

A. Analisis Kebutuhan

Penelitian dimulai dengan melakukan analisis kebutuhan perangkat lunak. Penulis mengumpulkan data mengenai kebutuhan perangkat lunak dengan menggunakan metode tanya jawab atau wawancara. Penulis melakukan wawancara dengan beberapa narasumber yaitu Koordinator Program Studi Rumpun Matematika FMIPA UNJ beserta admin dari program studi masing-masing, dan juga pengumpulan data yang diperoleh dari penelitian sebelumnya yang berjudul “*Monitoring Skripsi Mahasiswa Berdasarkan Kartu Bimbingan Skripsi Berbasis Web*”. [8]

B. Desain Sistem

Tahapan berikutnya yaitu tahap untuk membuat desain sistem. Dalam hal ini, penulis membuat desain dalam bentuk UML (*Unified Modelling Language*) antara lain:

1. *Use Case Diagram*

Use Case Diagram menunjukkan bagaimana *user* berinteraksi dengan sistem yang dikembangkan oleh penulis. *Use case diagram* yang dibuat menggunakan draw.io. Pada sistem ini terdapat empat aktor yaitu admin, koorprodi, dosen, dan mahasiswa yang digeneralisasi dari *user*. *use case diagram* dari sistem terdapat pada gambar berikut:

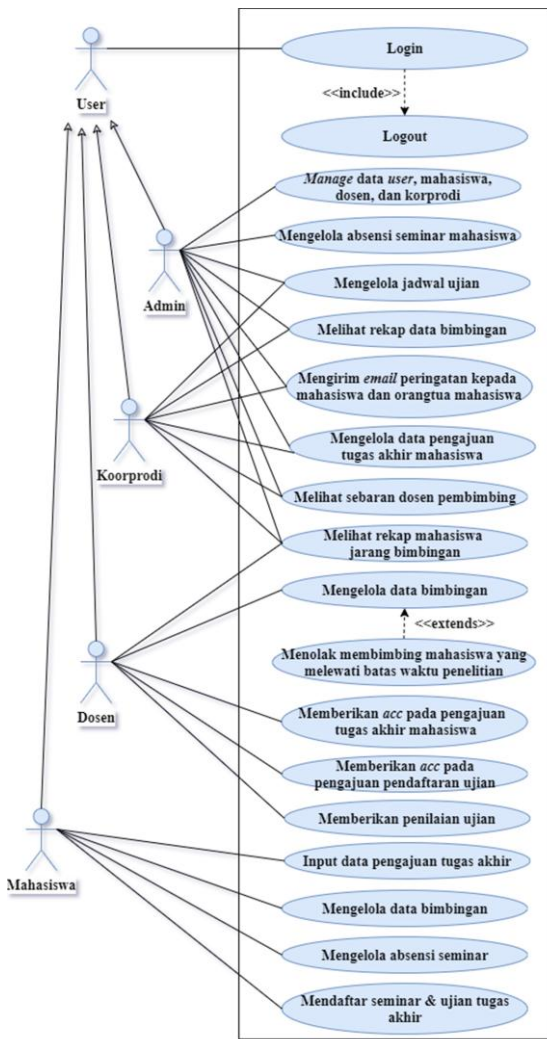
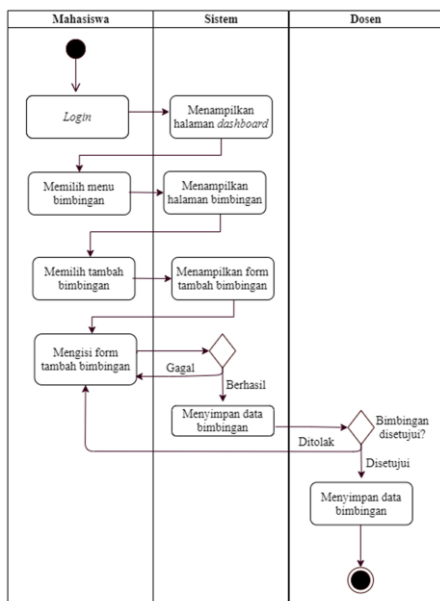


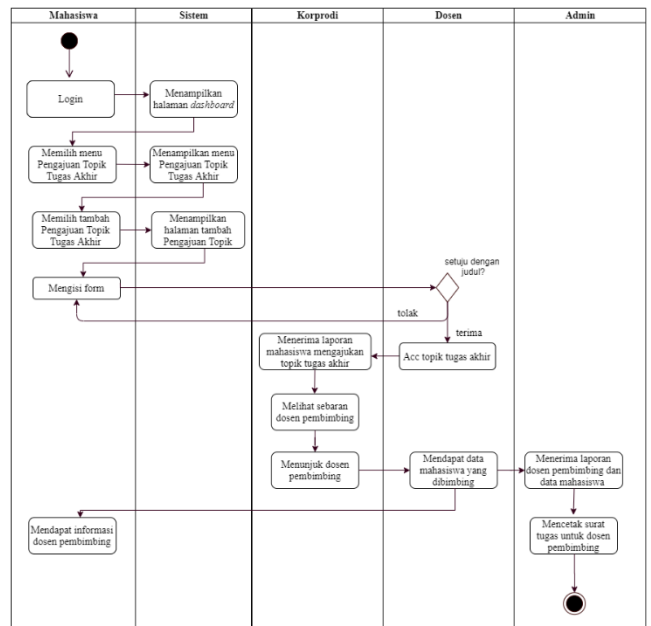
Diagram Use Case

2. Activity Diagram

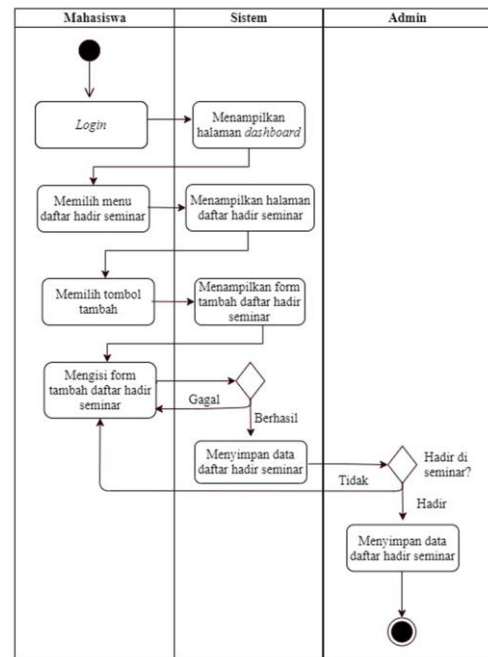
Pada sistem ini desain activity diagram sebagai berikut:



Activity Diagram Melakukan Bimbingan



Activity Diagram Pengajuan Judul Tugas Akhir



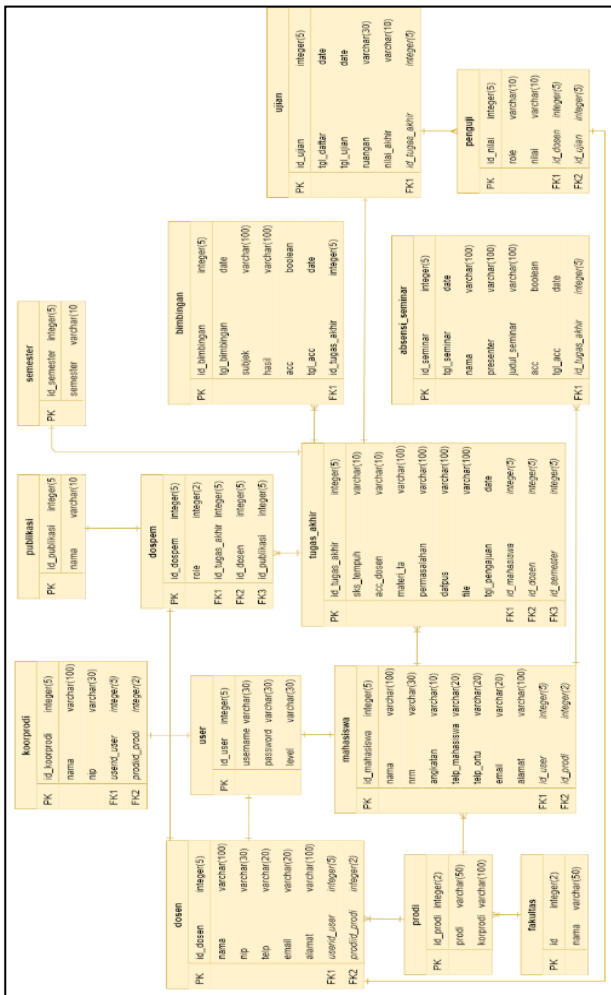
Activity Diagram Pendataan Kehadiran Mengikuti Seminar

3. Class Diagram

Pemodelan *Class Diagram* pada sistem ini menerapkan konsep MVC. *Model class* ditandai dengan warna abu-abu. Untuk *controller class* ditandai dengan warna biru. *Controller class* akan menghubungkan antara *model class* dan *view class*. *View class* ditandai dengan warna merah. Desain *class diagram* dapat dilihat pada gambar berikut:

5. Entity Relationship Diagram

Pemodelan ERD pada sistem ini terdapat pada gambar 3.20. Terdapat 17 (tujuh belas) entitas atau tabel yang masing-masing menyimpan data yang dibutuhkan dalam sistem.



ERD Sistem Informasi Sebaran Dosen Pembimbing

C. Pengembangan

Pada tahap ini adalah tahapan untuk mengubah semua desain kedalam kode program, berikut proses pengembangan pada sistem:

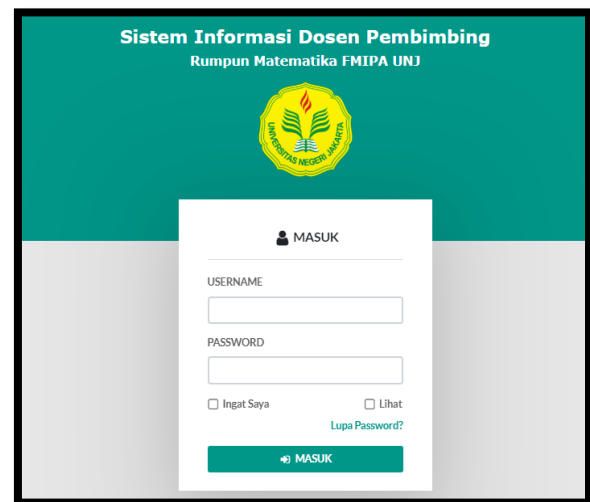
1. Membuat Database

Pada tahap ini dibuat *database* berdasarkan desain ERD yang telah dibuat pada desain sistem. *Database* dibuat menggunakan MySQL dan memanfaatkan aplikasi phpMyAdmin

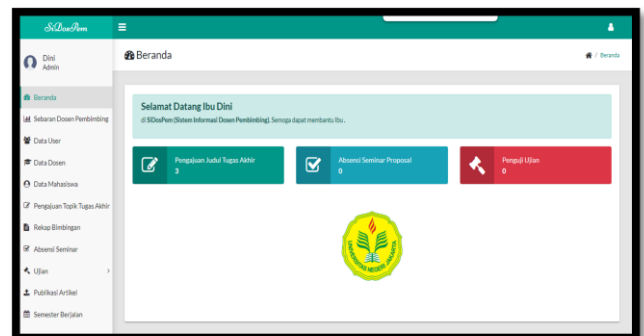
2. Implementasi Desain Tampilan

Pada tahap ini, digunakan Vali Admin *template* [9] yang menerapkan Bootstrap 4.0 dan PUG.js sebagai *framework* untuk membuat tampilan atau *front-end*. Berikut tampilan *website* Sistem Informasi Sebaran Dosen Pembimbing Rumpun

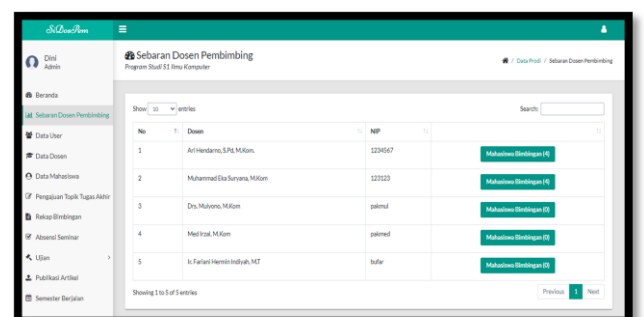
Matematika FMIPA UNJ yang dapat dilihat pada *browser*.



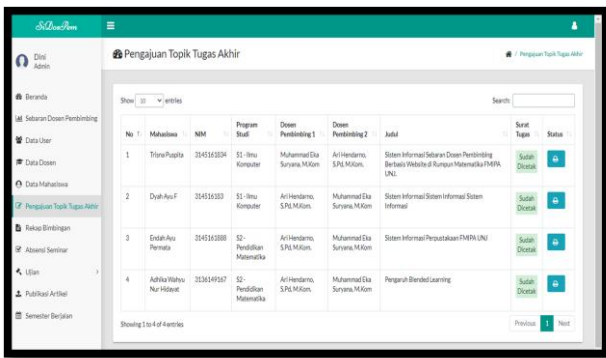
Tampilan Halaman Untuk Login



Tampilan Halaman Untuk Beranda



Tampilan Halaman Untuk Melihat Sebaran Dosen



Tampilan Halaman Pengajuan Tugas Akhir Pada Admin

3. Implementasi Sistem (*Back-End*)

Pada pengimplementasian semua fungsi yang berada dalam sistem, penulis menggunakan bahasa pemrograman PHP dan *framework* Laravel untuk memudahkan dalam mengimplementasikan arsitektur MVC.

IV. UJI COBA DAN HASIL UJI COBA

A. Uji Coba

Uji coba pada sistem dilakukan terhadap dua responden admin, tiga responden koorprodi, tiga responden dosen dan 12 responden alumni. Setiap responden melakukan pengujian terhadap sistem berdasarkan peran masing-masing responden. Uji coba yang dilakukan menggunakan data dari kuesioner Pengujian Penerimaan Pengguna atau yang disebut dengan *User Acceptance Test* (UAT).

Pada pengujian fungsional yang digunakan untuk penilaian seluruh komponen berdasarkan pilihan berikut:

- S : Setuju
- TS : Tidak Setuju

Untuk pengujian kebergunaan (*usability*) digunakan skala likert. Skala likert merupakan suatu skala penilaian yang menyajikan pilihan skala dengan nilai pada setiap skala untuk mengukur tingkat persetujuan terhadap sesuatu.[10] Skala yang digunakan dengan nilai 1 s.d. 5 dengan perincian sebagai berikut :

- 1 : Sangat Tidak Setuju
- 2 : Tidak Setuju
- 3 : Cukup
- 4 : Setuju
- 5 : Sangat Setuju

Setelah didapatkan seluruh nilai dari pengguna (*user*) saat pengujian, nilai tersebut dikalkulasi sesuai dengan sistem penilaian berikut:

- Nilai Total

Nilai total yaitu jumlah keseluruhan yang didapatkan dari setiap pertanyaan yang dapat ditulis menjadi:

$$\text{Nilai Total} = (\text{jumlah} \times \text{skorSS}) + (\text{jumlah} \times \text{skorS}) + (\text{jumlah} \times \text{skorC}) + (\text{jumlah} \times \text{skorTS}) + (\text{jumlah} \times \text{skorSTS})$$

- Presentase Kelayakan

Presentase kelayakan yaitu persentase nilai rata - rata yang didapatkan dari nilai total dibagi skor yang diharapkan. Skor yang diharapkan didapat dari skor maksimal yang dikalikan dengan jumlah responden. Skor maksimal adalah nilai maksimal dari skala likert dikalikan dengan jumlah pertanyaan. Perhitungan tersebut dapat ditulis menjadi:

$$\text{Presentase Kelayakan}(\%) = \frac{\text{NilaiTotal}}{\text{SkorDiharapkan}} \times 100\%$$

Persentase kelayakan yang telah didapatkan akan dibandingkan dengan skor pada skala likert. Berikut model skala likert:

- Sangat Kurang Sesuai = 0% - 20%
- Kurang Sesuai = 21% - 40%
- Cukup Sesuai = 41% - 60%
- Sesuai = 61% - 80%
- Sangat Sesuai = 81% - 100%

B. Hasil Uji Coba

Berdasarkan hasil pengujian fungsional dan kebergunaan pada semua *user*, didapatkan bahwa fitur-fitur yang terdapat pada sistem dapat berjalan dengan baik. Selain itu, didapatkan persentase kelayakan sebagai berikut:

- *User Admin* : 84,5%
- *User Koorprodi* : 81%
- *User Dosen* : 84%
- *User Mahasiswa* : 92%

Dari persentase masing-masing *user* kemudian dihitung total persentase kelayakan yang merupakan rata-rata dari nilai persentase kelayakan semua *user*, dapat dilihat sebagai berikut:

$$\text{Total Persentase Kelayakan}(\%) = \frac{84,5\% + 81\% + 84\% + 92\%}{4} = 85,4\%$$

Berdasarkan perhitungan tersebut didapatkan total persentase kelayakan senilai 85,4% berada pada

rentang tafsiran 81%-100%, maka dapat dikatakan bahwa nilai kebergunaan pada keseluruhan sistem mendapatkan predikat sangat sesuai.

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil implementasi dan pengujian program ini, maka didapat kesimpulan sebagai berikut:

1. Sistem Informasi Sebaran Dosen Pembimbing Skripsi dan Tesis di Rumpun Matematika FMIPA UNJ merupakan pengembangan dari penelitian Tiara Amelia (2017) dengan penambahan fitur dapat melihat sebaran dosen pembimbing, pencatatan absensi seminar dan penilaian ujian.
2. Pada awal proses pengembangan Sistem Informasi Sebaran Dosen Pembimbing Skripsi dan Tesis di Rumpun Matematika FMIPA UNJ menggunakan metode pengembangan perangkat lunak SDLC dengan Model Spiral namun karena terjadi kesalahpahaman sehingga SDLC yang digunakan berubah menjadi Model *Waterfall*.
3. Sistem Informasi Sebaran Dosen Pembimbing Skripsi dan Tesis di Rumpun Matematika FMIPA UNJ dikembangkan dengan bahasa pemrograman PHP dengan bantuan *framework* Laravel.
4. Berdasarkan hasil *User Acceptance Test* pada pengujian fungsional didapatkan bahwa fitur-fitur yang terdapat pada Sistem Informasi Sebaran Dosen Pembimbing Skripsi dan Tesis di Rumpun Matematika FMIPA UNJ dapat berjalan dengan baik dan sesuai kebutuhan.
5. Berdasarkan hasil *User Acceptance Test* pada pengujian kebergunaan (*usability*), didapatkan total persentase kelayakan dari keseluruhan sistem adalah 85,4%. Nilai tersebut terdapat pada skor skala likert 81%-100%, maka nilai kebergunaan Sistem Informasi Sebaran Dosen Pembimbing Skripsi dan Tesis di Rumpun Matematika FMIPA UNJ mendapat predikat sangat sesuai.

B. Saran

Adapun beberapa saran untuk penelitian selanjutnya adalah:

1. Meningkatkan ruang lingkup penelitian dari rumpun menjadi fakultas.
2. Sebaiknya tetap menggunakan metode pengembangan perangkat lunak SDLC dengan Model Spiral karena model ini dapat mengakomodasi kebutuhan yang berevolusi atau berubah terus menerus.
3. Mengintegrasikan Sistem Informasi Sebaran Dosen Pembimbing Skripsi dan Tesis di Rumpun Matematika FMIPA UNJ dengan Sistem Informasi Akademik Universitas Negeri Jakarta.

DAFTAR PUSTAKA

- [1] Hutahaen, J. (2014). Konsep Sistem Informasi. Yogyakarta: Deepublish Publisher.
- [2] Bartjya, S., Sharma, A., & Rani, U. (2017). A detailed study of Software Development Life Cycle (SDLC) Models. *International Journal Of Engineering And Computer Science*, 22097-22100.
- [3] Pressman, R. (2008). *Software engineering: a practitioner's approach seventh edition*. New York: McGrawHill.
- [4] Bentley, W. J., & Ld, D. (2004). *Metode Desain dan Analisis Sistem Edisi 6*. Yogyakarta: Andi.
- [5] Hendini, A. (2016). Pemodelan UML Sistem Informasi Monitoring Penjualan Dan Stok Barang. *Jurnal Khatulistiwa Informatika*, 107-116.
- [6] Miles, R., & Hamilton, K. (2006). *Learning UML 2.0*. Sebastopol: O'Reilly Media.
- [7] Yudhanto, Y., & Prasetyo, H. A. (2019). *Mudah Menguasai Framework Laravel*. Jakarta: PT. Elex Media Komputindo.
- [8] Amelia, T. (2017). *Monitoring Skripsi Mahasiswa Berdasarkan Kartu Bimbingan Skripsi Berbasis Web (Skripsi)*. Jakarta: Universitas Negeri Jakarta.
- [9] Borsadiya, P. (2020, Agustus 16). *Vali Admin*. Retrieved from Pratik Borsadiya's blog: <https://pratikborsadiya.in/blog/vali-admin/>
- [10] Maryuliana, S., & M. I., C. S. (2016). Sistem Informasi Angket Pengukuran Skala Kebutuhan Materi Pembelajaran Tambahan sebagai Pendukung. *Jurnal Transistor Elektro dan Informatika*.