

Pembuatan Aplikasi Sistem Portal Skripsi di Program Studi Teknik Informatika Universitas Kristen Petra Berbasis Website

Mahardhika Permana Putera¹, Liliana², Adi Wibowo³

Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Kristen Petra

Jl. Siwalankerto 121 – 131 Surabaya 60236

Telp. (031) – 2983455, Fax. (031) – 8417658

E-mail: mahardhikapp@gmail.com¹, lilian@petra.ac.id², adiw@petra.ac.id³

ABSTRAK

Program studi teknik informatika saat ini membutuhkan sistem yang dapat mengatur pendaftaran secara *online* dan dapat membantu untuk melakukan pengecekan jadwal dosen dan pencarian judul. Masalah lain yang pada program studi teknik informatika saat ini adalah pengaturan kuota dosen yang sering kali dapat melebihi batas atau kurang yang sangat jauh dari kuota yang ditentukan. Penjadwalan dan pengecekan jadwal dosen untuk sidang proposal skripsi maupun sidang skripsi juga merupakan masalah yang sedang dihadapi oleh program studi. Dengan adanya pengaturan kuota dalam aplikasi ini dapat membantu dosen dan mahasiswa sehingga pembagian mahasiswa dapat merata. Adapun pencarian judul untuk memudahkan mahasiswa melihat judul-judul skripsi lama sehingga mahasiswa tidak mengambil judul yang sama dengan yang sudah ada. Fitur pencarian judul ini menggunakan teknologi SOLR untuk melakukan proses pencarian yang dihitung berdasarkan kata dasar dari setiap kata yang dicari. Dalam proses pembuatan aplikasi ini digunakan *yii framework* dan *database* yang digunakan adalah *postgresql*. Berdasarkan hasil pengujian, aplikasi ini dapat berjalan dengan baik. Untuk ke depan mungkin dapat ditambahkan pengecekan judul skripsi yang sama secara struktur kata dan penawaran judul yang belum ada.

Kata Kunci: portal skripsi, *Yii Framework*, SOLR, *website*

ABSTRACT

At the moment, the Informatics Engineering Program needs a system that can manage an online registration, which can help in controlling the lecturers' schedules and also a better way for theses' title search. One problem that this department has, is in the system that controls the lecturers' teaching quota that is currently having an issue. The issue is the fact that a lecturer could teach way more than the quota and vice versa. The process of scheduling and checking of the lecturers' schedules for the theses' proposal and theses' defense is also a problem that raises in this department. However, the quota setting in this application will certainly could give an answer for both the lecturers and the students, since it will help making an equal division of the students' quantity for each lecturer. Moreover, the application also offers a better way for theses' title search in the sense that it makes the students easier to trace the previous theses' titles that already existed and avoid any overlapping titles in the future. This theses' title search feature uses SOLR technology to do a searching process that is counted based on the basic words from every searched word. The whole process in making this application is basically using yii framework and the

database is using postgresql. Based on the testing result, this application succeeding to run smoothly. For additional thing in the future, this application can be added with a feature to check the theses' titles that can no longer be used (already been used) and some title suggestions that are available (never been used yet).

Keywords: *theses portal, Yii Framework, SOLR, website*

1. PENDAHULUAN

Di zaman ini, teknologi informasi sudah semakin maju. Tidak dapat dipungkiri bahwa sistem informasi dewasa ini dibutuhkan untuk membantu pekerjaan sehingga menjadi lebih efektif dan efisien. Sistem informasi juga dapat mengefektifkan beberapa administrasi. Seperti saat sebuah perusahaan harus mengolah banyak data karyawan maka sistem informasi akan dapat membantu mengolah data karyawan dalam perusahaan tersebut secara teratur dan akan jauh lebih efektif daripada dilakukan secara manual tanpa menggunakan sistem informasi.

Di dalam Program Studi Teknik Informatika Universitas Kristen Petra proses pendaftaran skripsi sudah terkomputerisasi tetapi masih mengalami beberapa kendala. Dalam proses pengajuan topik, pengecekan kesamaan judul masih dilakukan secara manual dan dalam penginputan jadwal skripsi, belum ada pengecekan jika seorang dosen dijadwalkan di dua sidang pada saat yang bersamaan.

Sementara itu belum ada pencatatan pengajuan topik mahasiswa kepada dosen, hal ini menyebabkan masalah yaitu dosen tidak dapat mengecek kuotanya. Sehingga masih sering terjadi kesalahan dalam proses ini, seperti mahasiswa mengajukan topik kepada dosen yang sudah mencapai batas kuota yang telah ditentukan sehingga dosen menerima banyak anak bimbing. Masalah juga terjadi pada pencarian judul atau topik skripsi yang sudah pernah dibuat, mahasiswa tidak dapat melihat judul atau topik skripsi yang pernah dibuat sehingga masih dapat dijumpai mahasiswa memakai topik yang sama dengan topik yang sudah pernah ada.

Pembuatan aplikasi ini menggunakan *php* dengan *Yii framework*, untuk mesin pencarian judul skripsi digunakan teknologi SOLR. Dengan adanya aplikasi yang berbasis *website* dapat membantu mengurangi kendala-kendala yang terjadi pada saat proses pengambilan skripsi. Sehingga diharapkan proses pengambilan skripsi dapat dilakukan dengan efektif dan efisien.

2. LANDASAN TEORI

2.1 SOLR

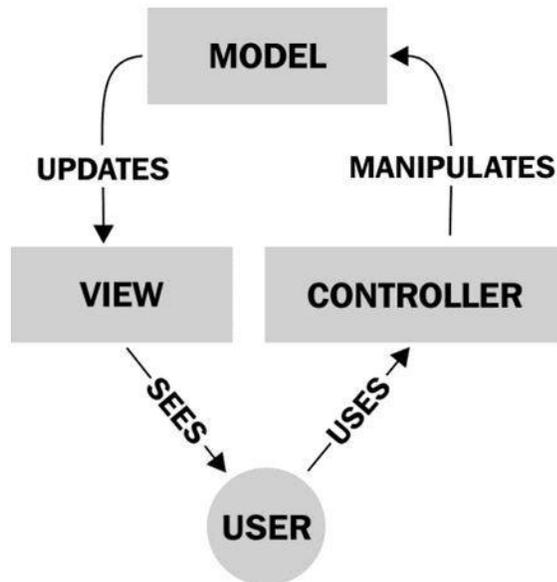
SOLR merupakan pengembangan dari Apache Lucene. Apache Lucene adalah *text search engine library* dengan performa yang tinggi. Apache Lucene pertama kali dikembangkan dan mulai dipublikasi secara bebas (*open sourced*) tahun 2000 dan dikembangkan oleh Doug Cutting. Sampai sekarang komunitas Lucene di internet masih sangat kuat. Saat ini *search engine* terbanyak menggunakan teknologi Lucene. Lucene merupakan sebuah *library* saja sehingga tidak bias berfungsi sebagai *server* dan bukan *web crawler* juga, tidak ada *file* konfigurasi juga di dalamnya. Untuk menggunakan Lucene dibutuhkan API untuk menulis *search code* yang sudah dibuat, dengan melakukan *indexing* terlebih dahulu untuk memasukkan dokumen ke dalamnya. Di dalam Lucene dokumen dapat disebut dengan *collection of fields*, yang di dalamnya terdapat teks atau angka. Kemudian melakukan pengaturan Lucene dengan *text analyzer* yang akan membentuk sebuah seri token (*tokenize*). Token yang terbentuk pada akhirnya akan disebut dengan *terms*. Proses akan dimulai dengan *analyzer*-nya yaitu *text analysis*. Kemudian Lucene mengindeks setiap dokumen di dalamnya, bisa disebut juga *index* yang tersimpan dalam *disk*. *Index* ini merupakan *inverted index* yang dimana menyimpan pemetaan dari setiap *field* yang ada pada *terms*. Setelah ini pencarian dapat dilakukan dengan *query string* yang di-*parsing* Lucene berdasarkan *syntax*-nya. Lucene memiliki beberapa fitur penting yaitu: *Inverted index* untuk efisiensi penerimaan dokumen melalui *indexed term*, memiliki set runtutan *text analysis component* seperti *tokenizers*, *query syntax* dengan *parser* dan berbagai jenis tipe *query* dari *simple term lookup* sampai *fuzzy matching*, *highlighter feature* untuk menampilkan kata yang ditemukan, *query spellchecker* berdasarkan isi dari *indexed* atau kamus yang disediakan, “*more like this*” untuk mencari *list* dokumen yang secara statistik mirip dengan yang dicari. Apache Solr merupakan *enterprise search server* yang menggunakan Lucene. Lucene merupakan bagian besar untuk mendefinisikan Solr. Untuk mengembangkan aplikasi dengan performa tinggi dan mempunyai banyak fitur dengan Lucene secara langsung sulit dan terbatas dengan aplikasi java. Untuk itu Solr membantu memecahkan masalah tersebut dengan memakai fitur-fitur dari Lucene dengan menggunakan *configuration files* dan parameter HTTP. Fitur utama dari Solr yang melebihi Lucene yaitu *server* yang dapat berkomunikasi melalui HTTP dengan menggunakan XML dan JSON *data format*, konfigurasi *file*, *cache* untuk mempercepat respon pencarian, *Administrative interface* berbasis *web*, *Faceting* hasil pencarian, dan *Geospatial search*. [6]

2.2 Yii Framework

Yii merupakan sebuah *open source framework*. Pertama kali Yii dikembangkan oleh Qiang Que pada bulan Januari 2008. Sebelum mengembangkan Yii, Qiang sudah pernah mengembangkan PRADO *framework* di waktu yang cukup lama. Dari pengalaman dan respon pengguna terhadapnya tersebut, Qiang memanfaatkannya untuk membuat *framework* yang jauh lebih mudah, lebih bisa dikembangkan, dan lebih efisien. Pada bulan Oktober 2008, Yii versi alpha dikeluarkan dan hasilnya sangat bagus dengan respon positif. Pada bulan Desember 2008, Yii versi 1.0 secara resmi dilepaskan ke publik, dan pada bulan Oktober 2012 versi 1.1.12 dilepaskan ke publik dan Yii terus dikembangkan setiap hari. [5]

Untuk perkembangan terakhirnya Yii telah mengeluarkan Yii versi 2.0 untuk versi beta pada bulan April 2013. Versi resmi yang telah keluar adalah Yii versi 1.1.14, yang dikeluarkan pada bulan Agustus 2013. Yii masih terus berkembang, tercatat dalam *web* resmi Yii sampai mempunyai lebih dari seribu jenis *extension*. Walaupun tidak semua *extension* terus dikembangkan tetapi masih ada yang dapat digunakan dengan baik. [3]

Yii adalah MVC [*framework*] dan menyediakan direktori yang terstruktur untuk kode *model*, *view*, dan *controller*, seperti terlihat pada Gambar 1. *Model* adalah bagian yang bertanggung jawab untuk menjaga sebuah *state* dan mencakup aturan bisnis yang diaplikasi ke data yang mendefinisikan sebuah *state*. *Model* di dalam Yii adalah setiap instansi dari *framework* kelas CModel atau turunan kelasnya. Di dalam *model* biasanya terdiri dari data atribut yang dapat memiliki label sendiri, dan dapat divalidasi berdasarkan aturan yang sudah dibuat di dalam *model*. Data yang membentuk atribut di dalam *model* bisa berasal dari baris dari tabel basis data atau dari *field* yang berada di *form input* untuk *user*. Yii mengimplementasikan dua jenis *model* yaitu *form model* (CFormModel) dan *active record* (CActiveRecord). Keduanya sama-sama berasal dari dasar yang sama yaitu kelas CModel. Untuk CFormModel adalah *model* yang mengumpulkan *form input* untuk HTML. Mencakup juga logika untuk validasi *form inputnya* dan logika bisnis lainnya yang mungkin dibutuhkan untuk *form input data*. Kemudian datanya dapat disimpan di dalam *memory* atau dengan bantuan *active record model* untuk menyimpan ke dalam basis data. Yang kedua *Active Record* adalah pola desain untuk akses basis data di dalam sebuah gaya penulisan *object-oriented*. Setiap objek *Active Record* di dalam Yii adalah instansi dari CActiveRecord atau turunan kelasnya, yang dimana mencakup satu baris di dalam tabel basis data termasuk semua logika dan rincian mengenai akses basis datanya dan logika bisnis yang dibutuhkan dan berhubungan dengan data. Nilai *data field* untuk setiap kolom di baris tabel merepresentasikan sebagai sifat dari *active record*. *View* bertanggung jawab untuk *rendering user interface*, sering kali berdasarkan data di dalam *model*. *View* di dalam Yii adalah sebuah *script PHP* yang berisi *user interface* berdasarkan elemen yang ada. Biasanya memakai HTML, tetapi juga dapat berisi kode PHP. Kode PHP yang ada di dalam *view* biasanya sangat *simple*, seperti penkondisian, iterasi atau hal-hal yang berhubungan dengan elemen Yii seperti HTML *helper class methods* atau *prebuilt widget*. Untuk logika lain yang lebih rumit harus dipisahkan dari *view* dan ditempatkan semestinya yaitu di *model* bila berhubungan langsung dengan data atau *controller* untuk yang logika lebih umum. *Controller* ada yang memiliki peran mengatur permintaan dan bertanggung jawab terhadap *input* dari pengguna, berinteraksi dengan *model* dan menginstruksikan *view* untuk *update* dan menampilkan secara benar. *Controller* di dalam Yii adalah sebuah instansi dari CController atau turunan dari kelas tersebut. Ketika *controller* berjalan, maka akan melakukan permintaan, dimana akan berinteraksi dengan *model* dan *me-render* tampilan yang benar. Sebuah *action* di dalam *form* yang paling *simple* adalah metode kelas *controller* yang nama awalnya diawali dengan *action*. [7]



Gambar 1. MVC Pada Yii

2.3 PostgreSQL

PostgreSQL adalah sistem *database* berbasis relasi objek yang *open source* dan memiliki arsitektur yang telah terbukti kuat dalam segi reliabilitas, integritas data dan *correctness*. [2] PostgreSQL mampu berjalan di atas berbagai sistem operasi, termasuk Linux, UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) dan juga Windows. PostgreSQL memiliki berbagai fitur yaitu *foreign keys*, *joins*, *views*, *triggers*, *stored procedures*, *Multi-Version Concurrency Control (MVCC)*, *point in time recovery*, *tablespaces*, *asynchronous replication*, *nested transactions (savepoints)*, *online/hot backups*, *a sophisticated query planner/optimizer*, dan *write ahead logging for fault tolerance*. PostgreSQL mendukung *character sets*, *multibyte character encodings* dan *unicode* yang berlaku secara internasional. [1]

Beberapa keuntungan penggunaan PostgreSQL sebagai *database* yakni : kekebalan terhadap *over-deployment*, dukungan yang lebih baik dari vendor lainnya, penghematan biaya pegawai yang signifikan, kehandalan dan stabilitas yang legendaris, *extensible*, menggunakan kode sumber yang disediakan, *cross platform*, dirancang untuk lingkungan dengan volume tinggi, *graphical user interface* untuk desain *database* dan *administration tools*. [4]

3. DESAIN SISTEM

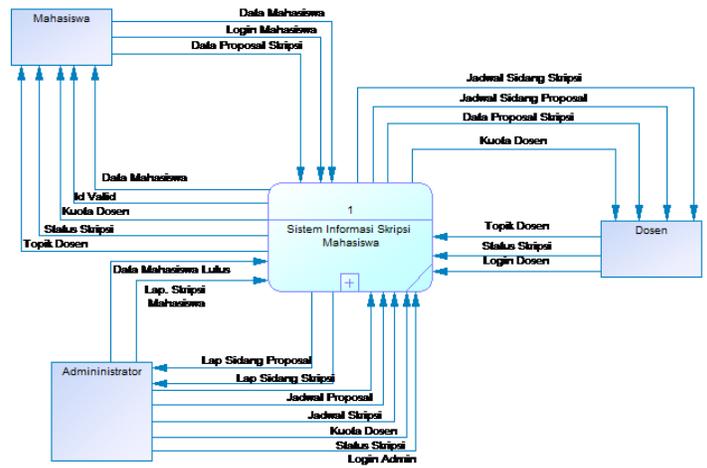
Dalam sistem, terdapat tiga jenis aktor yaitu mahasiswa dan dosen dan administrator. Untuk mahasiswa dan dosen adalah *user* yang terdaftar pada *account* pada salah satu *server* Petra (*john*, *peter*) dan *database* Program Studi Informatika Informatika. Administrator adalah koordinator proposal skripsi dan koordinator skripsi.

3.1 Data Flow Diagram

Data Flow Diagram (DFD) menjelaskan aliran proses keluar masuk pada Program Studi Teknik Informatika. DFD sistem informasi pendaftaran dibagi menjadi *Context Diagram*, dan DFD level 0.

3.1.1. Context Diagram

Context diagram merupakan gambaran umum dari proses pendaftaran skripsi pada Program Studi Teknik Informatika. Untuk lebih lengkap dapat dilihat pada Gambar 2.



Gambar 2. Context Diagram

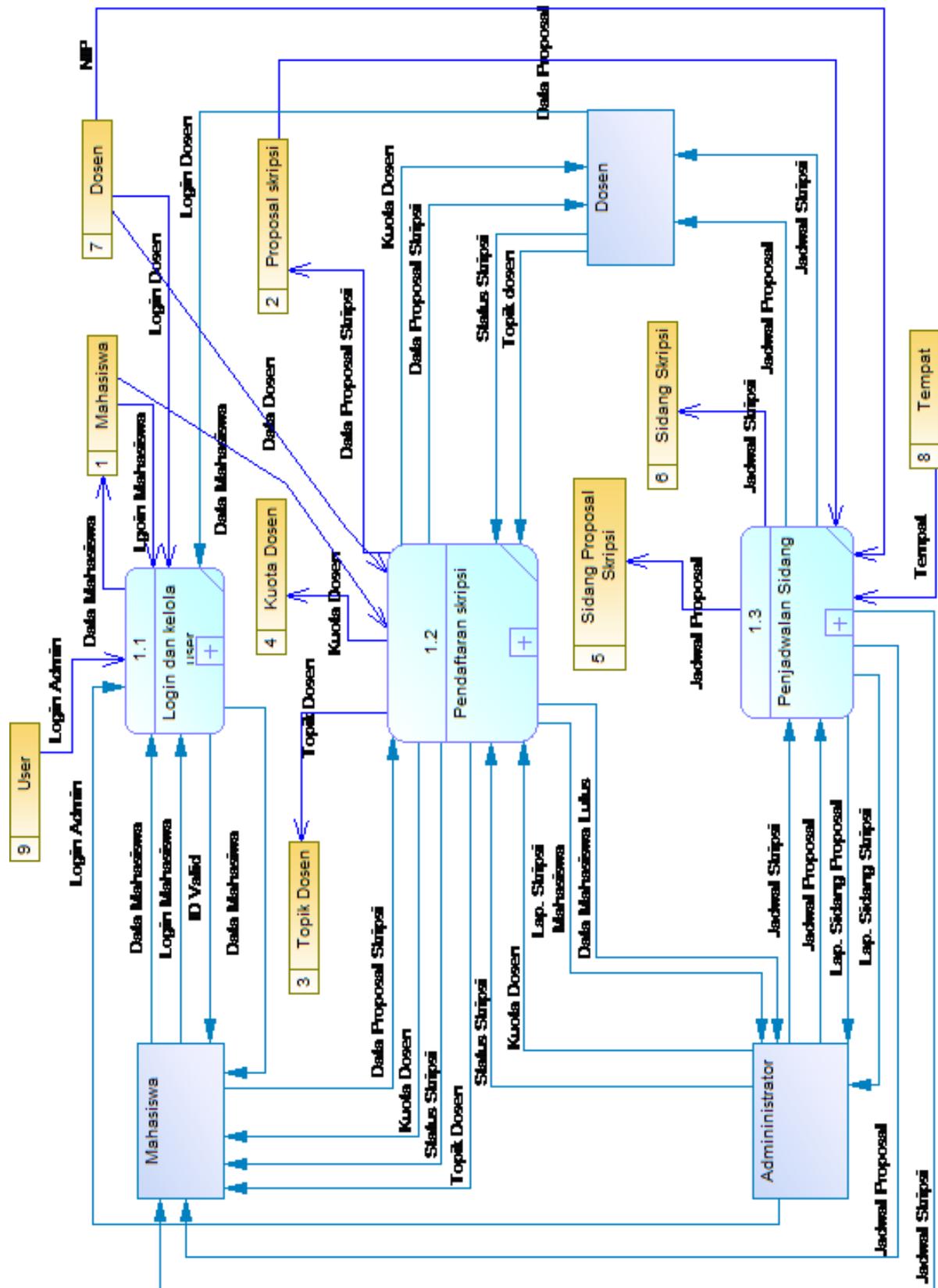
3.1.2. DFD Level 0

Pada DFD level 0 terdapat proses-proses utama yaitu proses *login*, proses pendaftaran skripsi, proses penjadwalan sidang, dan proses *maintain user*.

Proses *login* dan *maintain user* terjadi ketika mahasiswa, dosen dan administrator melakukan koneksi ke dalam sistem untuk mengakses sistem.

Proses pendaftaran skripsi terjadi ketika mahasiswa yang sudah terdaftar dalam tabel mahasiswa melakukan *entry data* untuk mengajukan permohonan skripsi. Atau waktu dosen menawarkan topik kepada mahasiswa. Selain itu juga saat administrator menetapkan kuota untuk masing-masing dosen, dan masih banyak lainnya.

Proses penjadwalan sidang terjadi ketika administrator meng-*entry* data jadwal sidang proposal skripsi maupun, data jadwal sidang skripsi. Untuk lebih lengkap dapat dilihat pada Gambar 3.



Gambar 3. DFD Level 0

3.2 Flowchart Sistem

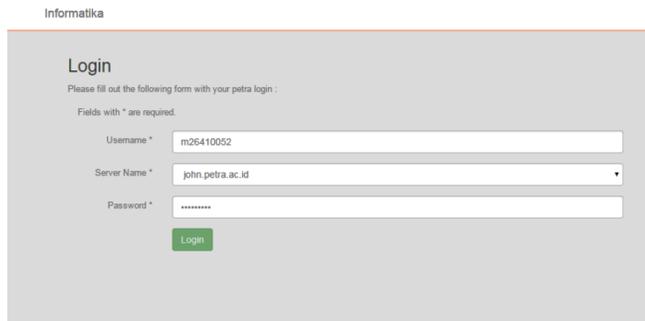
Flowchart merupakan sebuah gambar runtutan proses dari sistem pendaftaran skripsi saat ini. Dalam proses pendaftaran diawali dengan mahasiswa mengajukan topik kepada calon dosen pembimbing. Apabila calon dosen pembimbing tidak dapat melakukan pembimbingan karena kuota yang disediakan sudah penuh maka mahasiswa harus mencari calon dosen pembimbing yang lain. Setelah menjalani proses pencarian maka akan dilakukan sidang proposal. Apabila proposal diterima maka mahasiswa dapat mengerjakan skripsi untuk diajukan pada sidang skripsi. Untuk lebih jelasnya dapat dilihat pada Gambar 4.

3.3 Entity Relationship Diagram

Entity Relationship Diagram merupakan diagram yang menggambarkan entitas – entitas yang terdapat dalam sebuah sistem beserta relasi antar masing – masing entitas satu dengan yang lainnya. Entity Relationship Diagram juga merupakan dasar perancangan database dalam suatu sistem terutama dalam menentukan tabel – tabel yang harus dibuat beserta atribut-atributnya. Untuk lebih jelasnya dapat dilihat pada Gambar 5.

4. IMPLEMENTASI SISTEM

Aplikasi dimulai dengan menampilkan halaman login, seperti pada Gambar 6. Setelah user memasukan username dan password dengan benar maka user akan dialihkan sesuai dengan tipenya (mahasiswa, dosen, administrator). Untuk lebih lengkap bisa dilihat pada Gambar 6. Untuk mahasiswa terdapat fungsi utama yaitu input usulan, melihat kuota dosen, input proposal, dan mengirim email. Untuk halaman input proposal dapat pada Gambar 8. Dosen memiliki fungsi utama yaitu mengelola usulan yang masuk dan melihat jadwal sidang.



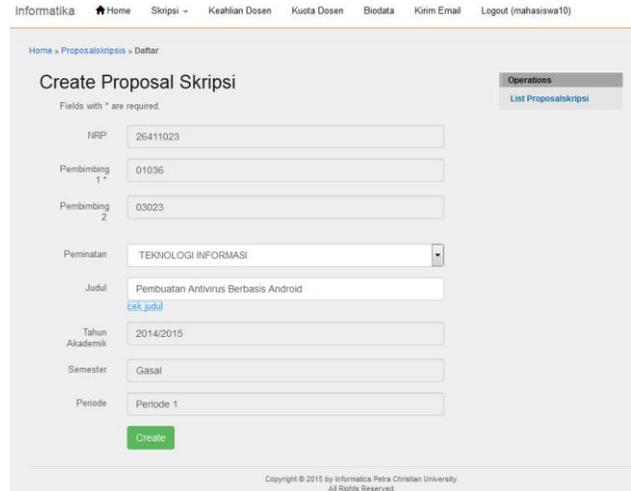
Gambar 6. Halaman Login

5. KESIMPULAN

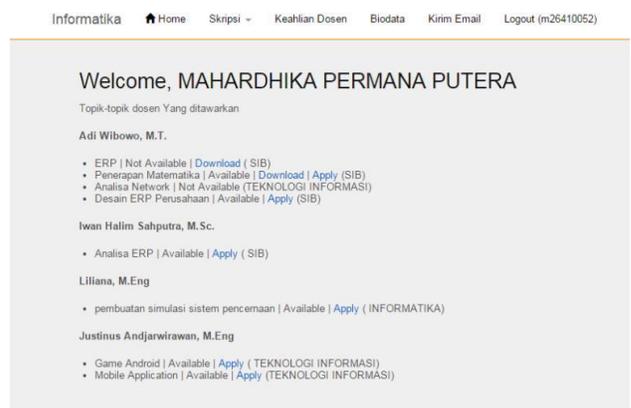
Berdasarkan hasil pengujian sistem, dapat disimpulkan beberapa hal sebagai berikut:

- Dapat berjalan sesuai dengan alur pada program studi teknik informatika.
- Dapat membantu memberikan solusi untuk masalah proses pendaftaran skripsi pada program studi teknik informatika.

Untuk ke depan mungkin dapat ditambahkan pengecekan judul skripsi yang sama secara struktur kata dan penawaran judul yang belum ada.



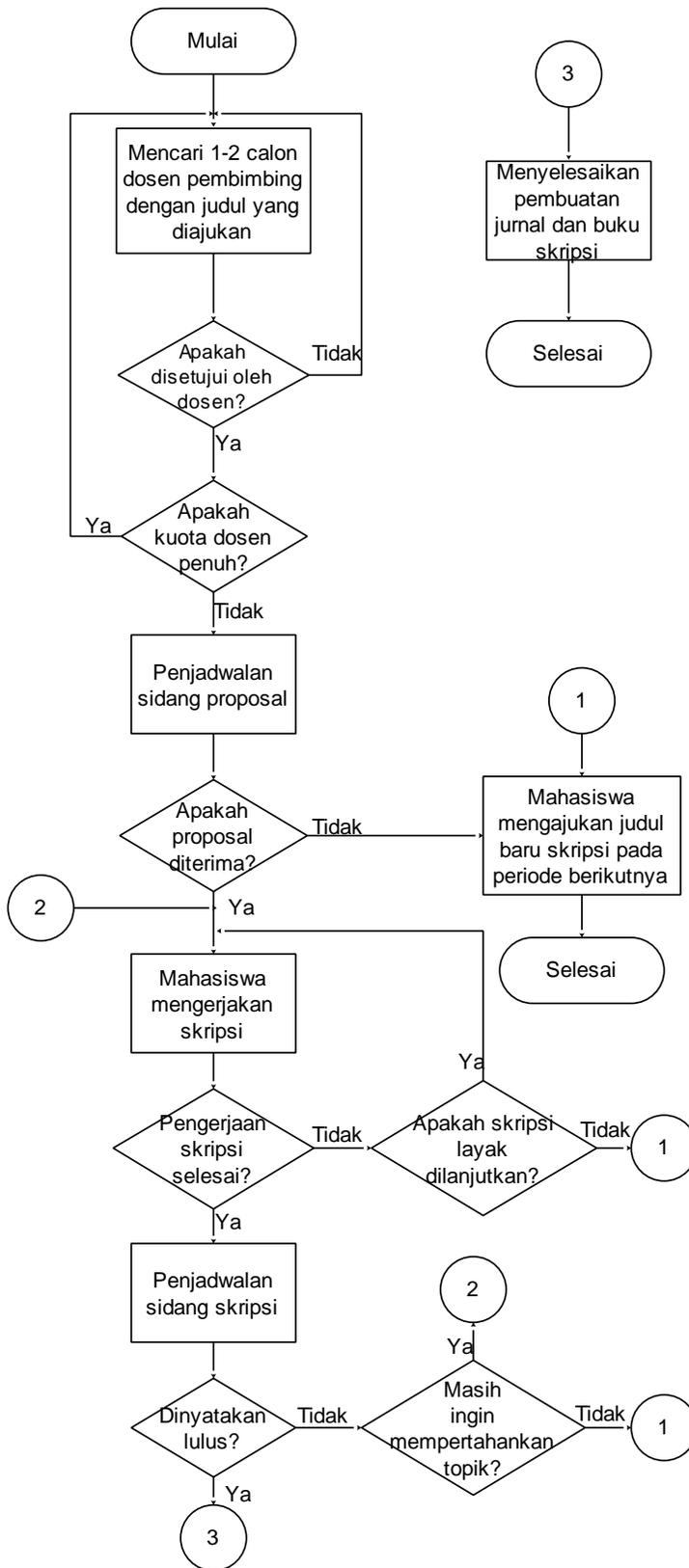
Gambar 7. Halaman Main Mahasiswa



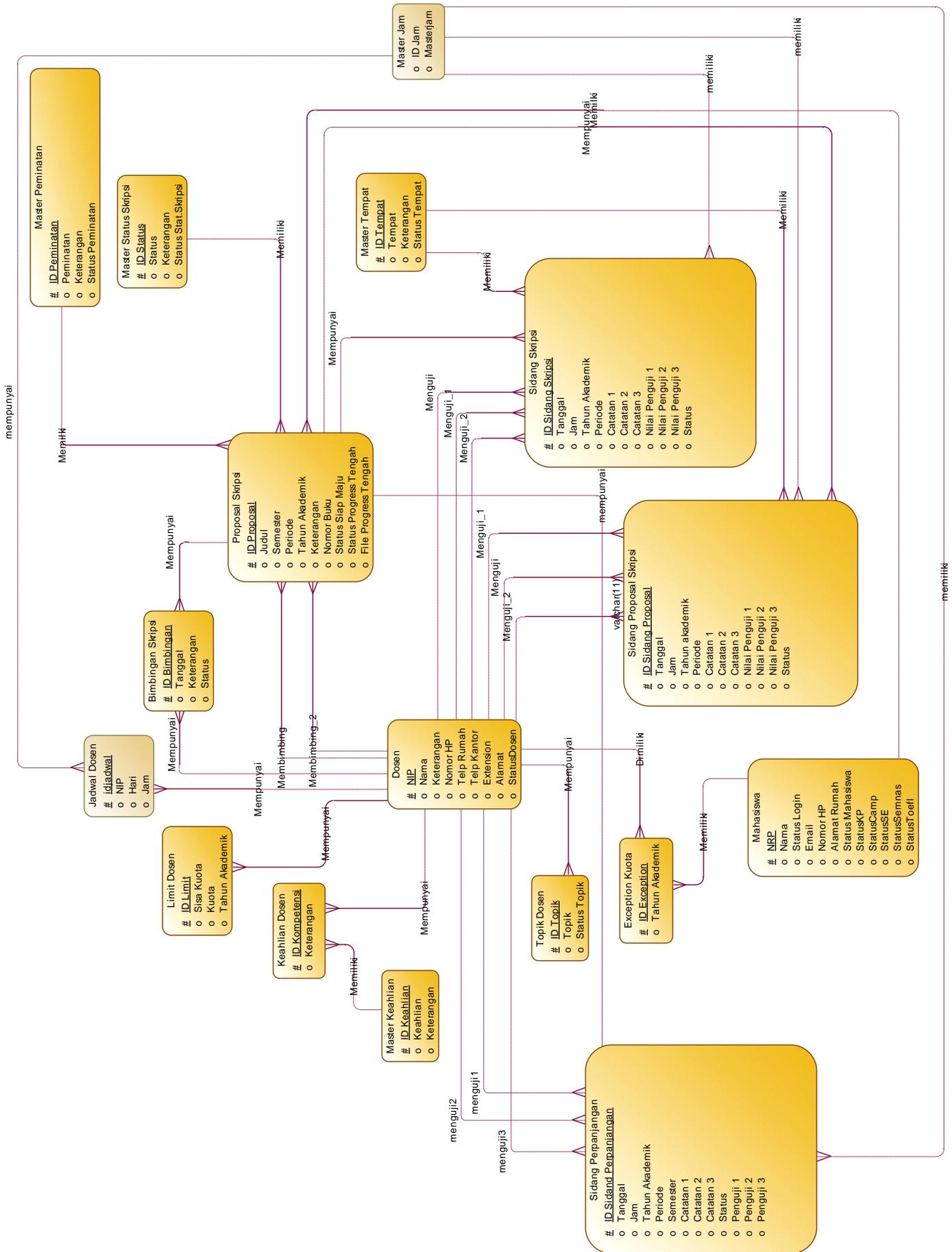
Gambar 8. Halaman Input Proposal

6. DAFTAR PUSTAKA

- [1] Douglas, K. dan Douglas, S. 2005. *PostgreSQL*, Second Edition. Sams.
- [2] Krosing, H., Mlodgenski, J., dan Roybal, K. 2013. *PostgreSQL Server*. Birmingham: Packt Publishing Ltd.
- [3] Makarov, A. 2013. *Yii Application Development Cookbook*. Birmingham: Packt Publishing Ltd.
- [4] Matthew, N. dan Stones, R. 2005. *Beginning Databases with PostgreSQL*. New York: Kinetic Publishing Services, LLC.
- [5] Mumm, J. dan Safronov, M. 2013. *Instant Yii 1.1 Application Development Starter*. Birmingham: Packt Publishing Ltd.
- [6] Smiley, D. dan Pugh, E. 2011. *Apache Solr 3 Enterprise Search Server*. Birmingham: Packt Publishing Ltd.
- [7] Winesett, J. 2012. *Web Application Development with Yii*. Birmingham: Packt Publishing Ltd.



Gambar 4. Flowchart Sistem



Gambar 5. Conceptual Diagram