

Analisis Desain Sistem Informasi Pengelolaan

Tugas Akhir Mahasiswa ISI Denpasar

Anak Agung Gede Bagus Ariana
Staff UPT. Teknologi Informasi dan Komunikasi
Institut Seni Indonesia Denpasar
gungariana@isi-dps.ac.id

Pelaksanaan Tugas Akhir bagi mahasiswa Institut Seni Indonesia (ISI) Denpasar merupakan salah satu syarat untuk kelulusan. Pelaksanaan tugas akhir yang diawali dengan beberapa proses administratif, seperti pendaftaran tugas akhir, pengumpulan berkas ujian proposal, pelaksanaan ujian proposal, pelaksanaan ujian akhir dan proses pengumpulan laporan akhir, saat ini berjalan dengan pencatatan pada aplikasi Microsoft Excel. Penggunaan Microsoft Excel sendiri belum terintegrasi dengan Sistem Informasi Akademik ISI Denpasar (JISTA) sehingga menyebabkan beberapa permasalahan. Permasalahan yang dihadapi antara lain, (1) belum rapinya administrasi registrasi dan pengumpulan berkas, (2) proses bimbingan belum termonitor dengan baik, (3) penerbitan SK berita acara ujian serta penilaian Tugas Akhir yang masih memerlukan beberapa proses pencetakan dokumen secara berulang. Permasalahan yang ditimbulkan tidak hanya pada proses administrasi data saja, permasalahan yang terkait dengan pengambilan keputusan seperti kuota pembimbing, proses pelaksanaan bimbingan tugas akhir yang kurang termonitoring, dan manajemen pencarian topik tugas akhir yang kurang praktis. Untuk mengatasi permasalahan tersebut, diperlukan suatu rancang bangun sistem informasi yang terintegrasi dengan sistem sebelumnya. Proses analisis sistem informasi dilakukan dengan melakukan wawancara terhadap staff akademik serta melakukan observasi terhadap pelaksanaan tugas akhir. Proses observasi tersebut, tentunya sangat tergantung dari proses analisis kebutuhan sistem. Proses ini dinamakan *requirements engineering*. Kegagalan pengembangan perangkat lunak terjadi akibat adanya kondisi tidak konsisten (*inconsisten*), tidak lengkap (*incomplete*), tidak benar (*incorrect*) dari *requirement specification*. Untuk mengatasi permasalahan tersebut, teknik analisis kebutuhan perangkat lunak atau *software requirement specification* harus dilakukan secara komprehensif. Adapun proses ini terdiri dari proses *requirement elicitation*, *requirement specification* dan *validation and verification*. Proses analisis kebutuhan yang tepat, tentunya dapat mengurangi kegagalan dalam pengembangan sistem informasi pengelolaan tugas akhir ini. Selain itu kebutuhan fungsional harus dapat diukur dan bersifat SMART yang terdiri dari *Specific* (spesifik), *Measurable* (terukur), *Achievable* (dapat dicapai), *Realistic* (masuk akal) dan *Traceble* (dapat dilacak). Adapun proses perancangan sistem menggunakan perangkat pemodelan Unified Modelling Language (UML), yang meliputi use case diagram dan activity diagram. Penggambaran dua diagram UML untuk menjelaskan ruang lingkup serta alur sistem secara lebih mudah oleh *user*, sehingga pemahaman mengenai sistem yang akan dibangun dapat dilakukan secara menyeluruh.

Kata kunci: Sistem Informasi, Tugas Akhir, Analisis Kebutuhan, SMART

1. Pendahuluan

Proses rancang bangun perangkat lunak merupakan gabungan antara proses manajerial dan juga proses teknis, untuk membangun, dan memelihara perangkat lunak secara sistematis. Proses ini merupakan bagian penting dari keseluruhan proses perancangan perangkat lunak [1][2]. Pada tahap ini, pengembang dan pengguna perangkat lunak saling berkomunikasi untuk dapat mewujudkan perangkat lunak yang dapat membantu pengguna mengatasi permasalahan-permasalahan yang mereka hadapi. Sehingga kemampuan manajerial serta kemampuan teknis sangat dibutuhkan oleh seorang sistem analis untuk menyelesaikan tugas ini dengan baik.

Kegagalan perangkat lunak sendiri, lebih banyak disebabkan oleh proses analisis kebutuhan yang kurang mampu mendefinisikan secara utuh kebutuhan-kebutuhan dari pengguna perangkat lunak [3]. Kegagalan ini terjadi akibat adanya kondisi tidak konsisten (*inconsisten*), tidak lengkap (*incomplete*), tidak benar (*incorrect*) dari *requerement specification*. Salah satu penyebab karena domain keahlian antara sistem analis dengan pengguna berbeda. Sehingga analis sistem harus mempelajari terlebih dahulu alur sistem yang dibutuhkan oleh pengguna. Selain itu terdapat permasalahan seperti kebutuhan yang terselubung, yang baru diketahui ketika proses perancangan perangkat lunak mulai berjalan. Hal ini dinamakan *undiscovered ruins syndrom*. Analisis sistem wajib mengeksplorasi keseluruhan kebutuhan baik yang sudah terlihat maupun yang belum terlihat. Sebab lainnya adalah *yes, but syndrome*, yang merupakan salah satu kendala yang sebenarnya mendeskripsikan bahwa, kebutuhan yang sudah digambarkan ternyata tidak mencerminkan kondisi yang riil diinginkan pengguna [4].

Salah satu kasus perancangan perangkat lunak yang akan diangkat pada Institut Seni Indonesia (ISI) Denpasar adalah Sistem Informasi Manajemen Tugas Akhir. Tugas akhir merupakan salah satu matakuliah yang ditempuh oleh mahasiswa ISI Denpasar yang merupakan salah satu syarat kelulusan. Tugas akhir sendiri memiliki rangkaian kegiatan yang melibatkan beberapa pengguna seperti mahasiswa, dosen pembimbing, operator program studi dan operator fakultas. Alur tugas akhir dimulai dari proses pendaftaran tugas akhir, pendaftaran proposal, ujian proposal, pendaftaran ujian akhir dan ujian akhir. Proses-proses ini saat ini berjalan secara manual, sedangkan sistem informasi akademik saat ini belum mengadopsi kebutuhan-kebutuhan pengguna untuk melakukan otomatisasi proses tugas akhir.

Mahasiswa mendaftar dengan sebelumnya berkonsultasi dengan Pembimbing Akademik (PA) untuk memvalidasi transkrip nilai, untuk memastikan jumlah SKS yang sudah ditempuh adalah 138 SKS. Sehingga mahasiswa hanya menyisakan matakuliah tugas akhir sebagai matakuliah terakhir mereka. Proses pendaftaran tugas akhir diawali dengan pengecekan

persyaratan pendaftaran yang terdiri dari transkrip nilai, nilai TOEFL, syarat aktif kegiatan organisasi mahasiswa serta pengecekan data pada PDDikti. Setelah syarat-syarat tersebut terpenuhi, mahasiswa melanjutkan dengan mengumpul laporan proposal mereka. Kondisi saat ini, mahasiswa mengunggah laporan mereka melalui Google Form. Selanjutnya operator program studi, mengirim nama-nama dosen pembimbing serta penguji tugas akhir kepada fakultas untuk dibuatkan Surat Keputusan (SK). Setelah SK ini diterbitkan, mahasiswa dapat melanjutkan proses bimbingan, untuk selanjutnya dijadwalkan untuk mengikuti ujian proposal tugas akhir. Ujian proposal tugas akhir ini, memberikan keputusan apakah judul yang diajukan oleh mahasiswa layak untuk dilanjutkan ke tahap selanjutnya, atau masih diperlukan revisi serta perbaikan-perbaikan. Tentunya keputusan serta diskusi yang terjadi saat pelaksanaan ujian proposal ini tertuang dalam berita acara pelaksanaan ujian proposal. Tahap akhir, setelah ujian proposal dan bimbingan berjalan, sampailah mahasiswa pada tahap ujian akhir. Proses yang sama dengan ujian proposal, mahasiswa mengumpulkan laporan akhir, dan menunggu jadwal ujian tugas akhir. Ujian tugas akhir ini pun menghasilkan berita acara ujian, yang nantinya digunakan sebagai dasar penilaian tugas akhir.

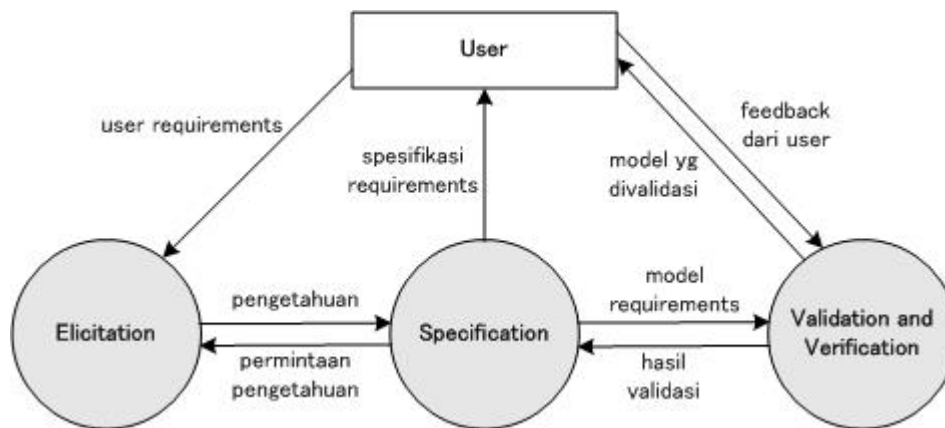
Keseluruhan proses tersebut belum sepenuhnya terkomputerisasi, sehingga menimbulkan beberapa permasalahan. Pihak prodi, fakultas dan dosen pembimbing kesulitan untuk melakukan monitoring proses penyelesaian tugas akhir oleh mahasiswa. Permasalahan selanjutnya, dokumentasi SK bimbingan oleh dosen pembimbing, belum tersimpan dengan baik. Terkadang dosen pembimbing meminta kembali SK bimbingan yang sebelumnya sudah diterbitkan. Proses validasi dan verifikasi dokumen syarat-syarat tugas akhir masih menggunakan kertas-kertas yang sebenarnya bisa dipangkas penggunaannya dengan validasi dan verifikasi secara komputerisasi.

Penelitian oleh Rahmi Fitri [3], menjabarkan penerapan SMART Requirement Analysis untuk mengurangi permasalahan-permasalahan yang terjadi akibat proses analisis kebutuhan sistem. Sistem yang dibangun oleh Rahmi adalah sistem informasi lomba karya tulis ilmiah yang mana kebutuhan perangkat lunaknya mengacu pada dalil *Specific* (spesifik), *Measurable* (terukur), *Achievable* (dapat dicapai), *Realistic* (masuk akal) dan *Traceble* (dapat dilacak). Hasil dari perancangan tersebut adalah dokumen spesifikasi perangkat lunak yang terdiri dari 13 list kebutuhan fungsional sistem.

Berdasarkan permasalahan-permasalahan yang telah diungkapkan maka pada penelitian ini akan dianalisis kebutuhan perangkat lunak dengan menggunakan SMART Requirement Analysis pada Rancangan Sistem Informais Tugas Akhir ISI Denpasar.

2. Metode

Metode yang digunakan pada mengacu pada gambar 1, yaitu *the three dimensions of requirements engineering*. Pada gambar terdapat beberapa aliran data seperti *user requirement* atau kebutuhan user, pengetahuan, permintaan pengetahuan, kebutuhan model, validasi hasil dari model, model yang divalidasi, umpan balik dari user dan kebutuhan spesifikasi.



Gambar 1. *The Three Dimensions of Requirements Engineering*

Penjelasan metode sebagai berikut:

1. Elicitation

Proses ini merupakan proses pengumpulan pengetahuan dari user. Pada tahap ini terjadi gap pengetahuan antara user dengan sistem analisi. Untuk mengatasi gap tersebut, digunakan teknik-teknik pemahaman pengetahuan dengan menggunakan (1) Story Board, (2) Interview, (3) Brainstorming, (4) Prototyping dan (5) Use case. [4]

2. Specification

Pengetahuan dari sistem yang diinginkan sebelumnya sudah terkumpul dan dapat dipahami dengan baik, langkah selanjutnya adalah membuat dokumen spesifikasi perangkat lunak. Proses penuangan pengetahuan dari proses pengumpulan pengetahuan sebelumnya memerlukan standarisasi agar nantinya dapat dipahami oleh banyak pihak. Standar dokumen yang digunakan adalah dokumen *IEEE Recommended Practice for Software Requirements Specifications* [IEEE-830]. Dokumen ini terdiri dari: *functional requirements*, *performance requirements*, *external interface requirements*, *design constraints*, dan *quality requirements*. Pada penelitian ini spesifikasi dokumen dituangkan dalam bentuk use case diagram, untuk mempermudah melihat actor yang terlibat serta *case* apa saja yang ada pada sistem. Proses penentuan *case* menggunakan pengelompokan list kebutuhan-kebutuhan yang ada.

3. Validation and Verification

Setelah dokumen spesifikasi perangkat lunak selesai, dilakukan kembali proses validasi dan verifikasi. Secara sederhana, validasi merupakan proses untuk memastikan bahwa kebutuhan perangkat lunak yang ditulis sudah benar, sedangkan verifikasi adalah proses untuk memastikan kebutuhan perangkat lunak sudah ditulis dengan benar. Proses validasi dan verifikasi ini melibatkan stekholder yang terlibat. Proses ini melibatkan dosen pembimbing, operator program studi, operator fakultas, serta mahasiswa.

3. Hasil dan Pembahasan

Tahap pertama sebelum melakukan proses analisis kebutuhan adalah penentuan actor yang terlibat pada sistem. Berikut adalah identifikasi actor pada sistem informasi pengelolaan tugas akhir:

Tabel 1. Identifikasi Aktor

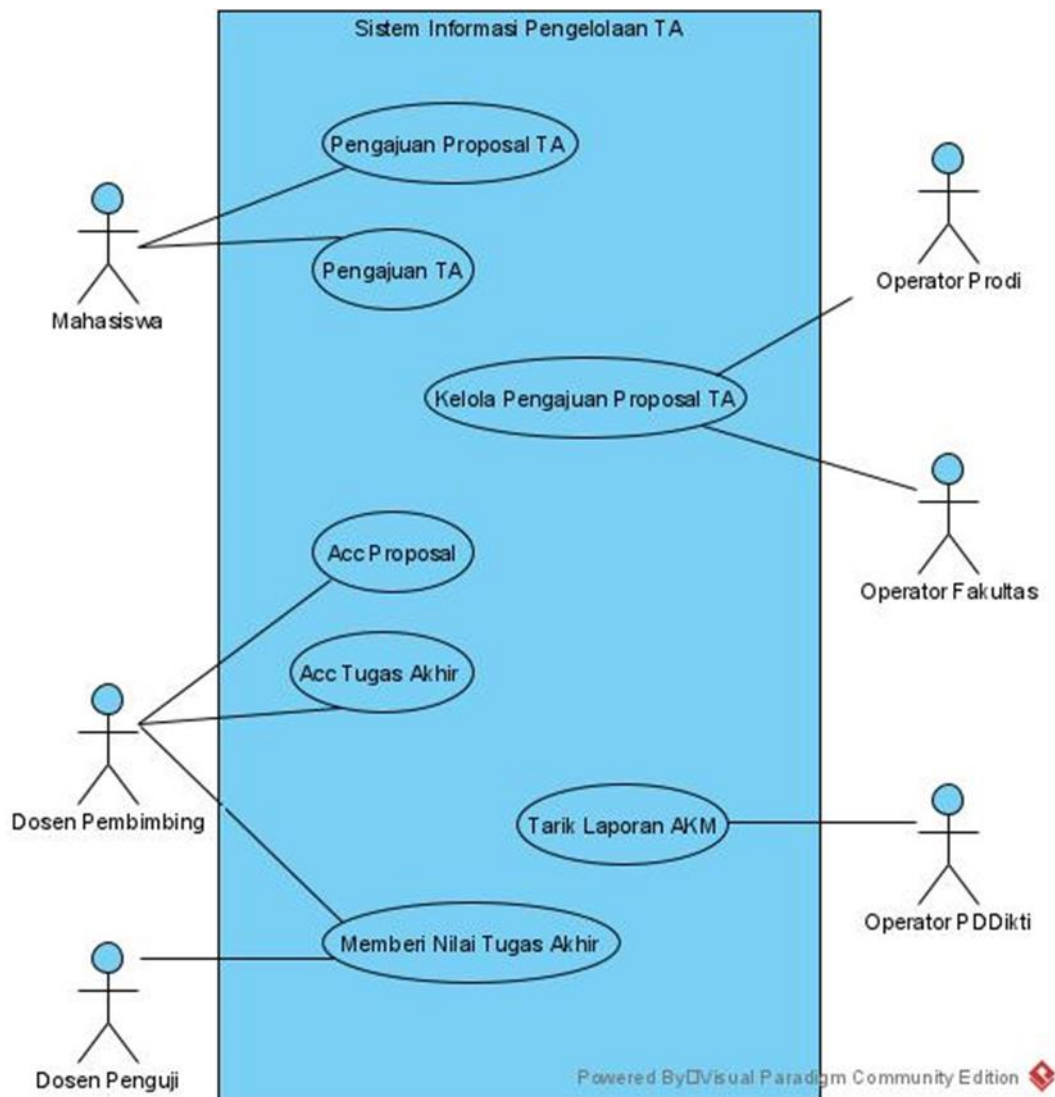
Aktor	Deskripsi
Mahasiswa	Dalam sistem mahasiswa dapat melakukan login ke sistem, menginput data proposal yang terdiri dari judul, topik dan latar belakang. Mengunggah dokumen pendukung syarat ujian proposal seperti dokumen keaktifan bidang kemahasiswaan.
Dosen Pembimbing	Dosen pembimbing dapat melihat mahasiswa yang dibimbing berdasarkan SK pembimbing yang telah diterbitkan. Dosen bisa memberikan feedback bimbingan pada sistem. Dosen dapat melakukan validasi terhadap ajuan mahasiswa untuk mengikuti ujian proposal dan ujian tugas akhir.
Operator Program Studi	Melakukan verifikasi data ajuan proposal dan tugas akhir mahasiswa. Membuat mapping pembimbing tugas akhir. Menentukan dosen penguji tugas akhir. Menentukan jadwal ujian proposal dan ujian tugas akhir.
Operator Fakultas	Melakukan verifikasi data mahasiswa dan pembimbing, serta membuat SK pembimbing dan penguji tugas akhir
Operator PDDikti	Menarik data aktivitas mahasiswa, yang terdiri dari nomor SK, judul tugas akhir, mahasiswa, dosen pembimbing, dan dosen penguji.

Selanjutnya adalah list kebutuhan fungsional yang diperoleh dari tahap *elicitation*. List kebutuhan ini dikelompokkan untuk menentukan *case* yang akan dimunculkan pada diagram *use case*. Pada tahap ini, masing-masing kebutuhan telah diverifikasi dengan mengacu pada terminologi SMART.

Tabel 2. Kebutuhan Fungsional Sistem

No.	Kebutuhan Fungsional	SMART
1	Mahasiswa menginput data pengajuan proposal	√
2	Mahasiswa mengunggah laporan proposal	√
3	Mahasiswa mengunggah bukti-bukti syarat tugas akhir	√
4	Menu Tugas Akhir akan muncul setelah KRS dengan matakuliah tugas akhir divalidasi oleh dosen pembimbing akademik	√
5	Mahasiswa dapat mengunggah laporan tugas akhir	√
6	Dosen pembimbing melakukan validasi terhadap laporan tugas akhir mahasiswa yang siap di ujikan	√
7	Dosen pembimbing mencetak SK dari sistem	√
8	Dosen pembimbing dan penguji memberi nilai kepada tugas akhir mahasiswa sesuai dengan form pada berita acara	√
9	Operator program studi mencetak jadwal ujian proposal serta ujian tugas akhir. File diunduh dari sistem dalam format Microsoft Excel	√
10	Operator fakultas melakukan verifikasi pembimbing dan penguji tugas akhir	√
11	Operator fakultas menerbitkan SK pembimbing dan penguji	√
12	Operator PDDikti mengunduh laporan dengan format Excel laporan aktivitas mahasiswa	√
13	Setelah terintegrasi dengan sistem akademik, sistem dapat memberikan notifikasi pengingat bahwa sedang mengambil matakuliah tugas akhir	√
14	Nilai tugas akhir merupakan rata-rata nilai pembimbing dan penguji, sistem dapat melakukan proses penghitungan nilai dan menghasilkan nilai akhir. Pembimbing dapat melakukan revisi terhadap nilai yang diberikan oleh sistem.	√
15	Sistem dapat menyimpan repository laporan tugas akhir yang sudah di setujui oleh pembimbing dan penguji. Repository disimpan tiap bab, untuk nantinya mempermudah publikasi ke sistem e-library.	√

Kebutuhan-kebutuhan fungsional yang sudah dijabarkan pada tabel 1, digambarkan dalam bentuk use case diagram untuk melihat interaksi antara sistem dan pengguna sistem. Use case merupakan abstraksi dari keseluruhan penjabaran kebutuhan fungsional.



Gambar 2. Use Case Diagram Sistem Pengelolaan Tugas Akhir

Terdapat enam aktor atau pengguna sistem serta tujuh *use case* yang merupakan pengelompokan dari kebutuhan fungsional yang sudah dijabarkan. Tahap selanjutnya setelah selesai melakukan proses desain sistem adalah validasi dan verifikasi desain sistem. Validasi dan verifikasi ini dilakukan melibatkan operator prodi, fakultas serta ketua program studi. Sistem akan dibangun berdasarkan modul-modul *use case* yang sudah dibangun. Penjabaran dari *use case* diagram adalah *narrative use case diagram*, *activity diagram* serta *sequence diagram*. Selanjutnya dari *sequence diagram*, sistem akan digambarkan dalam *class diagram* untuk mengetahui tabel-tabel apa saja yang akan dibangun pada *database* sistem. Langkah terakhir adalah membuat rancangan *user interface* sistem.

4. Penutup

Berdasarkan pemaparan diatas, dapat disimpulkan bahwa proses analisis kebutuhan merupakan proses yang penting dalam perancangan perangkat lunak. Diperlukan teori dan praktik untuk menjabarkan kebutuhan pengguna sehingga dapat menghindari kesalahan pembangunan perangkat lunak. Saran dan masukan untuk pengembangan selanjutnya adalah menggunakan model selain *SMART model requirement*, seperti *MOSCOW model requirement*.

Daftar Pustaka

- [1] Sadila, N. M., Pradana, F., & Priyambadha, B. (2017). Rekayasa Kebutuhan dengan Metode Pemodelan Berbasis Linguistik dan Ontologi pada Sistem Penilaian Prestasi Kerja Pegawai Dinas Kominfo Kota Malang. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 1(9), 765–773.
- [2] Rahimi Fitri, Subandi, Ida Hastuti. 2019. Analisis Rekayasa Kebutuhan Perangkat Lunak Menggunakan Smart Requirement Dalam Membangun Sistem Informasi Lomba Karya Tulis Ilmiah Politeknik Negeri Banjarmasin. Politeknik Negeri Banjarmasin
- [3] Hoffman, H., & Lehner, F. 2001. Requirements Engineering as a Success Factor in Software Projects. *IEEE Software*.
- [4] Romi Satria Wahono, Analyzing Requirements Engineering Problems, *IECI Japan Workshop 2003 (IJW-2003)*, Japan, 2003.