

RANCANG BANGUN SISTEM INFORMASI MENTORING DENGAN KONSEP SOFTWARE AS A SERVICE (SaaS)

DEVELOPING AN INFORMATION SYSTEM FOR RELIGIOUS MENTORING USING SOFTWARE AS A SERVICE (SAAS) CONCEPT

Yusep Rosmansyah¹, Fady Noor Ilmi²

Sekolah Teknik Elektro dan Informatika, Institut Teknologi Bandung
Jalan Ganesha 10, Bandung^{1,2}

yusep@stei.itb.ac.id¹, 18211015@std.stei.itb.ac.id²

ABSTRAK

Mentoring adalah bentuk pembimbingan dari mentor kepada *mentee* yang bertujuan untuk menghasilkan perkembangan positif dan sehat bagi *mentee*. Terbatasnya sumber daya menjadi masalah dalam pengelolaan *mentoring* di Indonesia karena lembaga pengelola *mentoring* pada umumnya merupakan lembaga nirlaba, khususnya lembaga masjid kampus dan sekolah. Dalam makalah ini, sebuah SI (sistem informasi) dengan konsep SaaS (*Software as a Service*) dirancang untuk mengatasi kendala keterbatasan sumber daya yang dimiliki oleh pengelola *mentoring*. Perancangan SI didasarkan pada analisis masalah yang dilakukan di *mentoring* Bidikmisi 2016/2017. Perancangan yang dilakukan terdiri atas perancangan arsitektur sistem, basis data, antarmuka, dan ketersediaan. Rancangan SI kemudian diimplementasikan dalam bentuk purwarupa *web app* dan mengalami pengujian fungsional dan pengujian penerimaan pengguna. Hasil pengujian fungsional menunjukkan bahwa semua kebutuhan fungsional berhasil dijawab melalui purwarupa dan rancangan sistem. Hasil pengujian penerimaan pengguna yang dilakukan kepada 46 pengguna dari pengelola, mentor, dan *mentee* menunjukkan bahwa SI berguna bagi pengelolaan *mentoring* (85%) dan mudah digunakan (57%).

Kata kunci: Sistem Informasi, Mentoring, SaaS, TAM 3, SUPR-Q

ABSTRACT

Mentoring is a form of guidance from mentors to mentees that aims to produce positive and healthy development for mentees. Limited resources are a problem in managing mentoring in Indonesia because mentoring management institutions are generally not-for-profit institutions, especially campus and school mosque institutions. In this article, an SI (information system) with the concept of SaaS (Software as a Service) is designed to overcome the constraints of limited resources of the mentoring manager. The design of the SI was based on the analysis of the problems carried out in Bidikmisi 2016/2017 mentoring program. The design consists of designing the system architecture, database, interface, and availability. The SI design was then implemented in the form of a web app prototype and undergoes functional testing and user acceptance testing. The functional test results showed that all functional requirements were successfully answered through prototyping and system design. The results of user acceptance tests conducted on 46 users consisting of managers, mentors, and mentees show that SI is useful for mentoring management (85%) and easy to use (57%).

Keywords: Information Systems, Mentoring, SaaS, TAM 3, SUPR-Q

PENDAHULUAN

Faktor utama kesuksesan sebuah bangsa adalah kualitas intelektual dan karakter sumber daya manusia yang dimilikinya. Salah satu metode dalam membina karakter adalah melalui *mentoring*. Penerapan *mentoring* di Indonesia terbukti

berpengaruh kuat terhadap peningkatan prestasi belajar dan perbaikan akhlak pemuda. Bahkan, proyek formalisasi kegiatan *mentoring* yang diselenggarakan di beberapa SMK di Jakarta pada tahun 2001-2004 menghasilkan penurunan

angka pelaku tawuran sebesar 77% (Dimiyathi, 2010).

Terdapat sejumlah tantangan dalam pengelolaan *mentoring* dan seiring dengan meningkatnya jumlah peserta, tantangan ini semakin mendesak untuk diatasi. Tantangan terbesar adalah sulitnya mengelola data-data kegiatan *mentoring* karena sistem informasi (SI) yang tersedia untuk mengelola semua data tersebut masih menuntut usaha yang besar. Pengelola *mentoring* tidak dapat mengadakan SI yang lebih baik karena minimnya dana dan sumber daya manusia yang mereka miliki. Bila tantangan ini tidak segera diatasi, pengelola akan kehabisan waktu dan tenaga, peningkatan kualitas mentor tidak dapat berjalan, dan pada akhirnya *mentee* akan merasa bosan dan berhenti dari kegiatan *mentoring*.

Seiring dengan perkembangan Internet, muncullah konsep *Software as a Service* (SaaS) yang membuat sebuah sistem informasi dapat digunakan oleh banyak pengguna (*multi-tenant*) dengan biaya yang rendah dan tanpa perlu pengadaan staf TI sama sekali. SaaS meminimalisasi biaya dengan membagi beban infrastruktur kepada setiap *tenant* dan staf TI tidak perlu diadakan oleh *tenant* karena pengelolaan infrastruktur TI sepenuhnya menjadi tanggung jawab penyedia layanan. Manfaat konsep SaaS sangat sesuai untuk mengatasi tantangan dalam pengelolaan kegiatan *mentoring* sehingga perlu adanya SI dengan konsep SaaS untuk pengelolaan *mentoring*. Bagian selanjutnya dari makalah ini akan mendiskusikan landasan teori,

metode, hasil dan diskusi, serta simpulan.

Mentoring

Istilah *mentoring* dapat didefinisikan sebagai bentuk hubungan antara pembimbing (mentor) dengan orang yang dibimbing (mentee) dengan pembimbing memberikan bimbingan, dukungan, dan semangat yang bertujuan untuk menghasilkan perkembangan positif dan sehat pada diri orang yang dibimbing (Bruce & Bridgeland, 2014). *Mentoring* merupakan model terbaik untuk pembinaan generasi muda karena adanya pemantauan yang lebih intensif terhadap perkembangan kualitas peserta *mentoring* dan terbangunnya ikatan yang lebih kokoh antarpeserta *mentoring*. Peran pengelola *mentoring* juga dibutuhkan untuk mengawal proses *mentoring*.

Mentoring dalam penelitian ini dimaksudkan pada *mentoring* yang diselenggarakan oleh Tim Pelaksana Program Pengembangan Karakter Beswan Bidikmisi (PBM) ITB. *Mentoring* Bidikmisi termasuk dalam jenis *mentoring* agama dan berkelompok. Sebuah kelompok *mentoring* Bidikmisi terdiri atas 6-10 orang mentee dan dipimpin oleh seorang mentor. Selain pertemuan pekanan, kegiatan pembinaan dalam *mentoring* Bidikmisi dilengkapi dengan acara pembinaan yang diselenggarakan satu kali setiap bulannya. Proses bisnis pengelolaan *mentoring* Bidikmisi lebih lengkap dapat dilihat pada Gambar 1 di bawah ini.



Gambar 1 Proses Bisnis Mentoring PBM ITB

Software as a Service (SaaS)

SaaS dideskripsikan sebagai model penyediaan perangkat lunak yang penyedia layanan memberikan lisensi atas sebuah aplikasi kepada konsumen untuk digunakan sebagai layanan berdasarkan kebutuhan (Zemin & Xiaofei, 2009). Penyedia layanan SaaS meletakkan aplikasinya di *web server* mereka sendiri atau mengunduh aplikasi tersebut ke perangkat konsumen. Setelah masa kontrak layanan habis, penyedia layanan menghentikan layanan untuk pengguna tersebut. Beberapa karakter utama SaaS adalah pembaharuan fitur terpusat, berarsitektur *single-instance* untuk *multi-tenant*, dikelola terpusat dan diakses lewat internet, penentuan harga berdasarkan jumlah pengguna, dan memiliki pendapatan dengan model langganan tanpa biaya lisensi awal (Ambulkar & Borkar, 2012). Aplikasi SaaS, yang disebut juga *cloud app*, berbeda dengan aplikasi web (*web app*) pada kemampuan untuk melayani klien yang sangat banyak (*multi-tenant*).

METODE

Secara keseluruhan, metode yang digunakan dalam penelitian ini mengacu pada metode Analisis dan Perancangan Sistem (Whitten & Bentley, 2007:160). Analisis sistem didefinisikan sebagai teknik pemecahan masalah yang menguraikan sebuah sistem menjadi potongan komponen-komponennya dengan tujuan untuk mempelajari seberapa baik setiap komponen bekerja dan berinteraksi satu sama lain untuk mencapai tujuannya. Analisis sistem dilakukan dengan lima tahapan, sebagai berikut.

Pendefinisian Lingkup

Pada tahap pendefinisian lingkup ditentukan masalah yang akan dipecahkan dan peluang yang ingin dijawab. Penentuan masalah dan peluang dilakukan berdasarkan jawaban pemilik sistem terhadap pertanyaan-pertanyaan dalam kerangka kerja PIECES. Masalah dan peluang yang ingin dipecahkan tercantum pada Tabel I.

TABEL I DAFTAR MASALAH DAN PELUANG HASIL

| No. | Masalah/Peluang | Klasifikasi |
|-----|--------------------------------|-------------|
| 1. | Data masukan tidak lengkap | Masalah |
| 2. | Data tidak terintegrasi | Masalah |
| 3. | Biaya cukup tinggi | Masalah |
| 4. | Proses dilakukan secara manual | Masalah |
| 5. | Kontrol terlalu ketat | Masalah |
| 6. | Sistem sulit dipelajari | Masalah |

| | | |
|----|--|---------|
| 7. | Akses layanan via aplikasi <i>Mobile</i> | Peluang |
| 8. | Layanan berbagi materi untuk mentor | Peluang |

Analisis Masalah

Pada tahap analisis masalah dilakukan eksplorasi akar masalah terhadap masalah dan peluang yang telah ditentukan pada tahap sebelumnya. Eksplorasi dilakukan dengan menggunakan diagram Ishikawa, atau yang biasa disebut diagram tulang ikan (*fishbone*

diagram). Klasifikasi penyebab masalah yang digunakan terdiri atas faktor manusia (*people*), faktor operasional (*procedure*), faktor teknologi (*plant*), dan faktor kebijakan (*policy*). Daftar hasil analisis masalah dicantumkan pada Tabel II.

TABEL II DAFTAR MASALAH DAN PELUANG HASIL ANALISIS MASALAH

| No. | Masalah/Peluang | Klasifikasi |
|-----|--|-------------|
| 1. | Data tidak terekam | Masalah |
| 2. | Teknologi yang ada tidak memberikan notifikasi pengingat | Masalah |
| 3 | Operator (pengelola) tidak menguasai layanan yang digunakan | Masalah |
| 4 | Teknologi yang ada tidak dapat membatasi otoritas pengguna | Masalah |
| 5 | Teknologi yang ada tidak dapat mengintegrasikan data antar file secara otomatis | Masalah |
| 6 | Teknologi yang ada tidak dapat menghasilkan beberapa informasi secara otomatis | Masalah |
| 7 | Teknologi yang ada tidak dapat diintegrasikan dengan layanan pengirim pesan (email atau messenger) | Peluang |
| 8 | Akses layanan via aplikasi <i>Mobile</i> Layanan berbagi materi untuk mentor | Peluang |

Analisis Kebutuhan

Sistem yang dituju harus dapat menjawab akar masalah yang diidentifikasi pada tahap sebelumnya. Pada tahap ini dilakukan penentuan kebutuhan yang dapat menjawab akar masalah tersebut. Kebutuhan-

kebutuhan ini dikelompokkan menjadi kebutuhan fungsional dan nonfungsional. Daftar kebutuhan fungsional ditampilkan pada Tabel III dan daftar kebutuhan nonfungsional ditampilkan pada Tabel IV.

TABEL III DAFTAR KEBUTUHAN FUNGSIONAL SISTEM

| ID | Deskripsi |
|-----------|---|
| SRS-F-001 | Sistem dapat merekam data penting untuk registrasi pengguna |
| SRS-F-002 | Sistem dapat membatasi otoritas antar peran (pengelola, mentor, dan mentee) dan antar-pengguna. |
| SRS-F-003 | Sistem dapat melakukan backup data. |
| SRS-F-004 | Pengelola dapat mengelola data mentor dan mentee yang tergabung dalam komunitasnya. |
| SRS-F-005 | Pengguna dapat memperbarui data profilnya masing-masing. |
| SRS-F-006 | Pengelola dapat mengelola data kelompok dalam komunitasnya. |
| SRS-F-007 | Mentor dapat mengelola data kehadiran peserta mentoring di kelompoknya. |
| SRS-F-008 | Mentee dapat melihat data kehadiran kelompoknya. |
| SRS-F-009 | Pengelola dapat mengelola data kehadiran peserta mentoring setiap kelompok di komunitasnya. |
| SRS-F-010 | Sistem memberikan laporan hasil mentoring kepada tiap-Tiap pengguna. |
| SRS-F-011 | Pengelola dapat mengelola data acara untuk komunitasnya. |
| SRS-F-012 | Sistem dapat mempublikasikan program acara berdasarkan target peserta yang sesuai. |
| SRS-F-013 | Sistem memiliki konfirmasi kehadiran dalam program acara. |
| SRS-F-014 | Sistem memberikan bukti kehadiran peserta acara bagi pengelola dan peserta. |
| SRS-F-015 | Sistem memiliki fitur analytics untuk memenuhi kebutuhan proses evaluasi. |
| SRS-F-016 | Pengelola dapat mengelola data formulir evaluasi untuk komunitasnya. |
| SRS-F-017 | Sistem dapat mengumumkan formulir evaluasi kepada target yang sesuai. |
| SRS-F-018 | Sistem memiliki mekanisme bagi mentor dan mentee Untuk mengisi formulir evaluasi. |
| SRS-F-019 | Sistem memfasilitasi pengelola dan mentor untuk saling berbagi materi pembinaan kepada komunitasnya. |
| SRS-F-020 | Sistem memiliki mekanisme bagi pengelola untuk menyetujui terlebih dahulu materi pembinaan yang diajukan oleh mentor. |
| SRS-F-021 | Sistem dapat memberikan notifikasi sebagai pengingat dan pemberitahuan atas kejadian penting terkait pengguna. |
| SRS-F-022 | Sistem diintegrasikan dengan layanan pengiriman pesan, seperti email atau Line untuk mengirimkan notifikasi. |
| SRS-F-023 | Sistem menyimpan data setiap tenant dalam skema yang sama |

TABEL IV DAFTAR KEBUTUHAN NONFUNGSIONAL SISTEM

| ID | Deskripsi |
|------------|--|
| SRS-NF-001 | Sistem memiliki antarmuka yang mudah digunakan dan dipelajari oleh pengguna |
| SRS-NF-002 | Sistem dapat diakses dengan mudah lewat perangkat mobile |
| SRS-NF-003 | Grafik-grafik dimasukkan ke file lain. |
| SRS-NF-004 | Sistem memiliki tampilan yang nyaman digunakan pada berbagai jenis perangkat |
| SRS-NF-005 | Sistem dapat melakukan pengolahan data dengan cepat |
| SRS-NF-006 | Sistem dapat menampilkan data dengan cepat |
| SRS-NF-007 | Sistem dapat menjaga kerahasiaan data penting |

Rancangan Logis

Pada tahap rancangan logis dilakukan pendokumentasian kebutuhan bisnis menggunakan model sistem yang mengilustrasikan struktur data, proses bisnis, aliran data, dan antarmuka. Ilustrasi ini membantu dalam validasi kebutuhan yang dihasilkan dari fase sebelumnya.

Analisis Keputusan

Pada tahap analisis keputusan dilakukan identifikasi kandidat-kandidat solusi yang dapat memenuhi kebutuhan tersebut, menganalisis tiap-tiap kandidat, lalu memberikan rekomendasi target sistem yang dituju. Analisis kandidat solusi dilakukan dengan menggunakan matriks kelayakan (*feasibility matrix*) berdasarkan empat kriteria, yaitu segi teknis, operasional, ekonomi, dan jadwal.

Terdapat dua kandidat solusi yang dapat memenuhi kebutuhan sistem di atas, yaitu sistem dengan model pengiriman layanan *software on demand* dan sistem dengan model pengiriman layanan menggunakan *hosted application management (hosted AM)*. Solusi dengan pengiriman layanan *software on demand* menggunakan satu salinan aplikasi yang dibuat secara spesifik untuk pendistribusian SaaS berupa *widget* dan *website* penyedia layanan. Solusi dengan model pengiriman layanan *hosted AM* menggunakan *website* penyedia layanan yang selalu tersedia. Analisis kelayakan terhadap kedua kandidat solusi berupa *feasibility matrix* di Tabel 5 menunjukkan bahwa solusi SI dengan model pengiriman layanan berupa *hosted AM* lebih unggul dan patut dipilih menjadi solusi yang akan diimplementasikan

TABEL V PERBANDINGAN MODEL PENGIRIMAN LAYANAN *SOFTWARE ON DEMAND* DENGAN MODEL PENGIRIMAN LAYANAN *HOSTED AM*

| Kriteria | Bobot | Solusi SI dengan Model Pengiriman Layanan <i>Software on demand</i> | Solusi SI dengan Model Pengiriman Layanan <i>Hosted AM</i> |
|--------------------------------|--------------|---|---|
| <i>Economic Feasibility</i> | 30% | Biaya implementasi = Rp 2.000.000,00 Biaya Operasional per semester = Rp 2.730.000,00 | Biaya implementasi = Rp 0,00 Biaya operasional per semester = Rp 2.500.000,00 |
| | | Nilai: 50 | Nilai: 90 |
| <i>Technical Feasibility</i> | 30% | Penyedia layanan Bertanggung jawab atas Pemeliharaan dan pengelolaan layanan. Pengelola mentoring Bertanggungjawab atas Komputer dan jaringan internet anggotanya, serta <i>website</i> utama komunitas mentoring. | Penyedia layanan Bertanggung jawab atas Pemeliharaan dan pengelolaan layanan. Pengelola mentoring Bertanggung jawab atas Komputer dan jaringan internet anggotanya. |
| | | Nilai: 70 | Nilai: 90 |
| <i>Operational Feasibility</i> | 30% | Solusi ini dapat menjawab masalah pada sistem. Data akan terintegrasi dan dapat dilakukan pengolahan data sehingga menghasilkan informasi baru. Dengan solusi ini, pengguna sistem dapat mengakses <i>website</i> utama komunitas mentoring untuk menggunakan fitur-fitur transaksional bagi mentee. Namun, untuk mengakses fitur-fitur manajerial, pengguna harus mengakses <i>website</i> penyedia layanan. | Solusi ini dapat menjawab masalah dan peluang pada sistem. Data akan terintegrasi dan dapat dilakukan pengolahan data sehingga menghasilkan informasi baru. Solusi ini dapat menghadirkan semua fitur Pada satu <i>website</i> sehingga pengguna sistem dapat melakukan tugasnya Tanpa perlu mengakses <i>website</i> lain. |
| | | Nilai: 80 | Nilai: 90 |

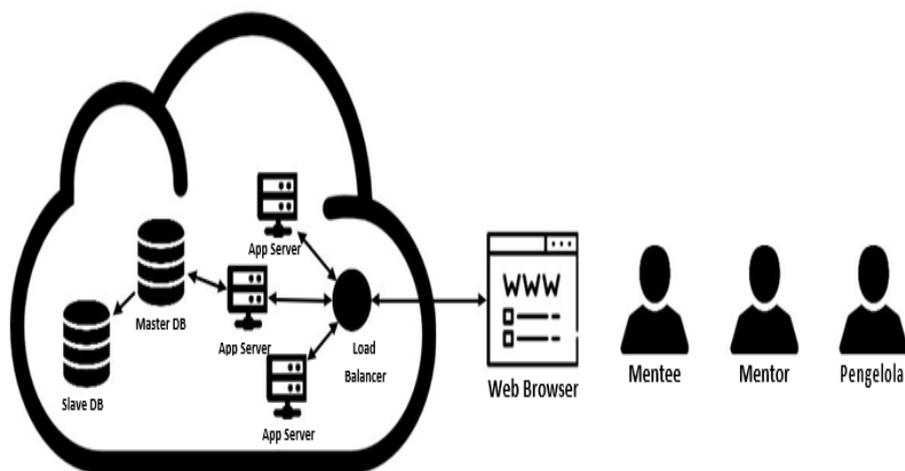
| | | | |
|-----------------------------|------|---|---|
| <i>Schedule Feasibility</i> | 10% | Dua belas pekan (3 pekan untuk analisis dan perancangan, 7 Pekan untuk pembangunan SI utama, 1 pekan untuk membangun layanan berupa <i>widget</i> , dan 1 pekan untuk instalasi). | Sebelas pekan (3 pekan untuk analisis dan perancangan, 7 Pekan Untuk pembangunan SI, dan 1 pekan untuk instalasi) |
| | | Nilai: 85 | Nilai: 90 |
| Ranking | 100% | 69 | 90 |

Setelah analisis sistem dilakukan untuk mengidentifikasi masalah yang dialami bisnis, berikutnya adalah merancang sistem dari segi teknis. Perancangan sistem didefinisikan sebagai sekumpulan tugas yang fokus pada spesifikasi rinci dari solusi berbasis komputer. Perancangan sistem terdiri atas tiga tahapan berikut.

Perancangan Arsitektur Aplikasi

Pada tahap perancangan arsitektur aplikasi ditentukan teknologi yang akan digunakan untuk mengimplementasikan sistem.

Penentuan setiap komponen aplikasi didistribusikan ke setiap lokasi bisnis juga termasuk dalam tahap ini. Sebagai pemenuhan karakteristik SaaS berupa pengelolaan perangkat lunak yang terpusat, maka SI dirancang untuk dapat diimplementasikan di *cloud* dan terdiri atas *Load Balancer* (LB), *Application Server*, dan dua jenis basis data: Master DB sebagai basis data utama dan Slave DB sebagai basis data cadangan. Rancangan arsitektur sistem ditampilkan pada Gambar 2.

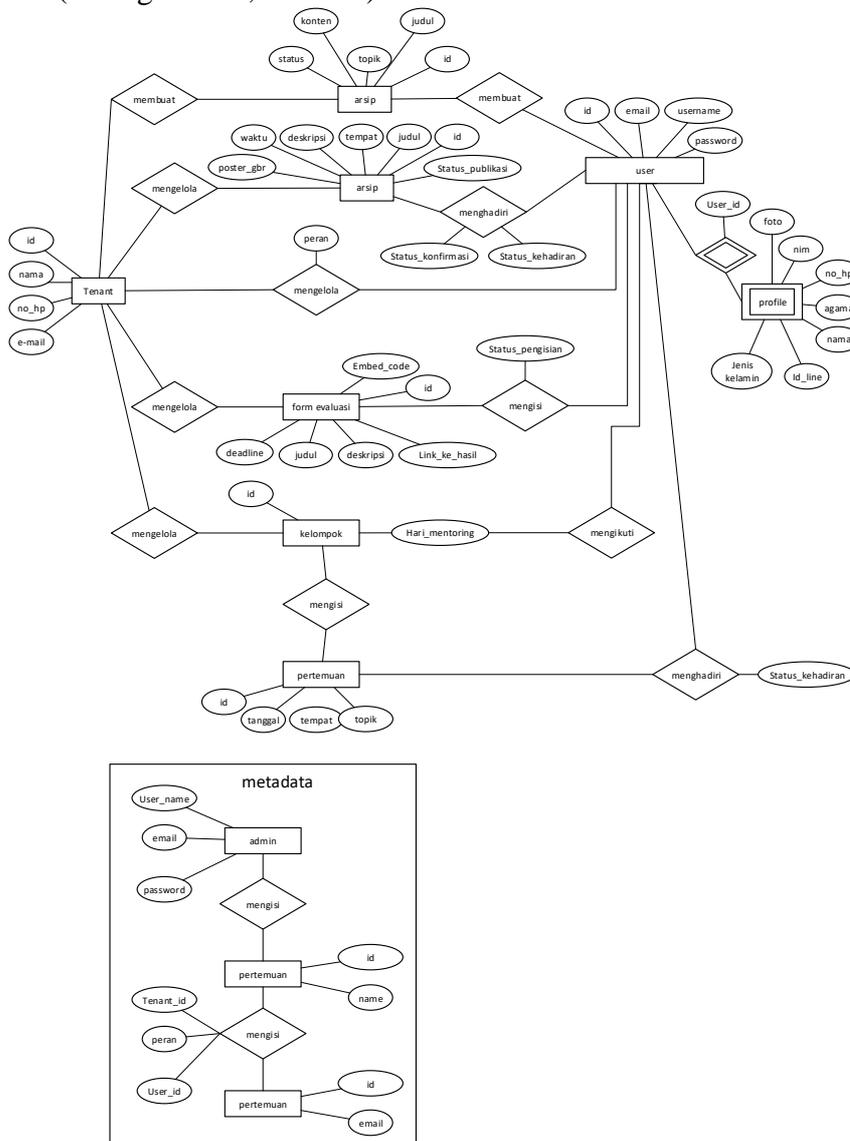


Gambar 2 Rancangan Arsitektur Aplikasi

Perancangan Basis Data

Pada tahap perancangan basis data digunakan *Entity Relationship Diagram* untuk merepresentasikan struktur data dari sistem yang dituju. Basis data yang dirancang diharapkan dapat beradaptasi terhadap kebutuhan yang akan datang dan pengembangan sistem. Terdapat tiga pendekatan basis data untuk aplikasi *multi-tenant*, yaitu basis data terpisah, basis data bersama dengan skema terpisah, dan basis data bersama dengan skema bersama (Chong dkk, 2006).

Pendekatan basis data yang digunakan untuk SI yang dirancang ini adalah basis data dengan skema bersama. Pendekatan ini dipilih karena lembaga pengelola *mentoring* yang bersifat nirlaba membutuhkan sistem dengan biaya pengelolaan yang murah dan secara umum proses bisnis antar- pengelola *mentoring* seragam tanpa membutuhkan penyesuaian. Rancangan basis data sistem dapat dilihat pada Gambar 3.

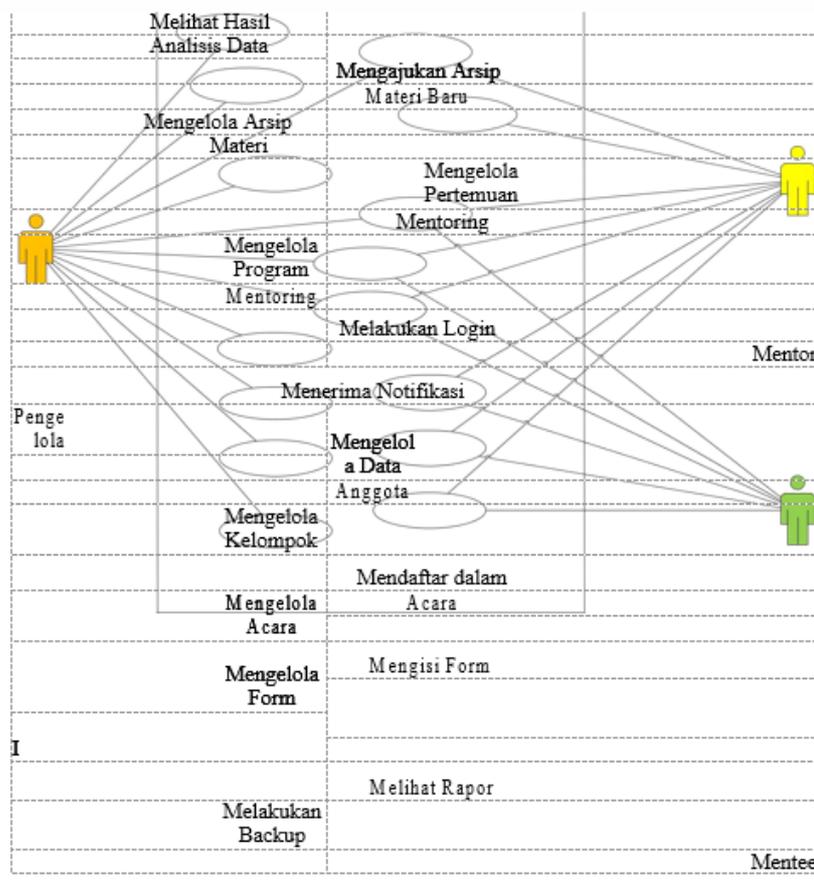


Gambar 3 Rancangan Basis Data Sistem

Perancangan Antarmuka

Pada tahap perancangan antarmuka dilakukan pengembangan spesifikasi input, output, dan dialog yang sesuai untuk pengguna sistem. *Use Case Diagram* digunakan untuk memberikan gambaran besar interaksi antara sistem dengan

pengguna. Terdapat tiga jenis aktor yang akan berinteraksi dengan SI ini, yaitu pengelola *mentoring*, mentor, dan *mentee*. Interaksi antara aktor dengan SI digambarkan dalam bentuk *Use Case Diagram* seperti pada gambar 4.



Gambar 4 Rancangan *Use Case Diagram* Sistem

Perancangan Ketersediaan

Salah satu karakteristik SaaS adalah ketersediaannya yang sangat tinggi. Ketersediaan (*availability*) merupakan salah satu bagian paling penting dari jaminan suatu layanan (Hunnebeck, 2011:125). Tidak tersedianya layanan menyebabkan utilitas layanan tidak dapat diakses sehingga bisnis tidak akan merasakan

nilai yang dijanjikan. Dengan demikian, perancangan SI yang menerapkan SaaS perlu mencakup perancangan ketersediaan layanan.

Perancangan ketersediaan dilakukan dengan membagi ketersediaan layanan ke dalam dua kelompok yaitu fungsionalitas operasional dan fungsionalitas nonoperasional. Fungsionalitas

operasional merupakan fungsionalitas SI yang berkaitan dengan kegiatan operasional *mentoring*, yang terdiri atas pengelolaan data anggota, pengelolaan data kelompok, dan pengelolaan data kehadiran dalam pertemuan *mentoring* dan acara pembinaan. Fungsionalitas

nonoperasional merupakan fungsionalitas lain yang tidak berkaitan dengan kegiatan operasional seperti berbagi arsip meteri dan pengelolaan formulir evaluasi. Rancangan ketersediaan kedua fungsionalitas dapat dilihat pada table-table berikut

TABEL VI RANCANGAN KETERSEDIAAN FUNGSIONALITAS OPERASIONAL SISTEM

| | | |
|--|--|---|
| Waktu Layanan | Setiap hari, pukul 08.00 – 21.00 WIB | |
| Kebutuhan Ketersediaan | Jumlah gangguan yang diperbolehkan | 2 jam / pekan |
| | Ambang batas ketersediaan | 97,80% |
| | <i>Downtime</i> untuk perawatan | Tidak ada (perawatan dilakukan di luar waktu layanan) |
| | Prosedur untuk menginformasikan adanya gangguan terhadap layanan | Gangguan terencana diinformasikan kepada pengelola <i>mentoring</i> melalui e-mail dan dilakukan paling lambat 30 menit sebelum terjadinya gangguan tersebut. |
| | | Gangguan tak terencana diinformasikan kepada pengelola <i>mentoring</i> melalui e-mail dan dilakukan paling lambat 30 menit setelah gangguan terjadi. |
| Kebutuhan kinerja | Kapasitas yang dibutuhkan layanan | 30 transaksi/detik |
| | <i>Response time</i> dari aplikasi | 100 – 10.000 ms |
| | Waktu reaksi dan resolusi | Reaksi dilakukan dalam waktu maksimal 10 menit setelah insiden. Resolusi dilakukan paling lambat dalam waktu 30 menit setelah insiden |
| Kebutuhan untuk melakukan perawatan bila terjadi insiden | Perawatan dilakukan pada 00.00 – 01.00, 1 hari setelah terjadinya insiden. | |

TABEL VII RANCANGAN KETERSEDIAAN NON-FUNGSIONALITAS SISTEM

| | | |
|------------------------|------------------------------------|-------------|
| Waktu Layanan | 7 hari 24 jam | |
| Kebutuhan Ketersediaan | Jumlah gangguan yang diperbolehkan | 2 jam/pekan |
| | Ambang batas ketersediaan | 96.43 % |
| | <i>Downtime</i> untuk perawatan | 4 jam/pekan |

| | | |
|--|--|--|
| | Prosedur untuk menginformasikan adanya gangguan terhadap layanan | Gangguan terencana diinformasikan kepada pengelola mentoring melalui e-mail dan dilakukan paling lambat 30 menit sebelum terjadinya gangguan tersebut. |
| | | Gangguan tak terencana diinformasikan kepada pengelola mentoring melalui e-mail dan dilakukan paling lambat 30 menit setelah gangguan terjadi. |
| Kebutuhan kinerja | Kapasitas yang dibutuhkan layanan | 100 transaksi/detik |
| | <i>Response time</i> dari aplikasi | 100 – 10.000 ms |
| | Waktu reaksi dan resolusi | Reaksi dilakukan dalam waktu maksimal 10 menit setelah insiden. Resolusi dilakukan paling lambat dalam waktu 30 menit setelah insiden |
| Kebutuhan untuk melakukan perawatan bila terjadi insiden | | Perawatan dilakukan pada 00.00 – 01.00, 1 hari setelah terjadinya insiden. |

Selanjutnya, agar hasil rancangan sistem dapat dinilai kesesuaiannya dengan kebutuhan, dilakukan implementasi hasil rancangan sistem dalam bentuk purwarupa dan pengujian terhadapnya. Purwarupa yang dibuat berupa aplikasi web. Pengujian yang dilakukan adalah pengujian fungsional dan uji penerimaan pengguna (*user acceptance test*). Dalam uji penerimaan pengguna, *Techonology Acceptance Model 3* (TAM 3) (Venkatesh & Bala, 2008) menjadi dasar dalam pengujian kegunaan (*perceived of usefulness*) sistem dan *Standardized User Experience Percentile Rank Questionnaire* (SUPR-Q) (Sauro, 2015) menjadi dasar dalam pengujian tingkat kemudahan (*perceived of ease of use*) sistem untuk digunakan.

HASIL DAN PEMBAHASAN

Sistem diimplementasi dalam bentuk *web app*. *Web app* yang dibuat terdiri atas 45 antarmuka berdasarkan

rancangan antarmuka *web app* yang dirancang. *Web app* diimplementasikan dengan menggunakan *framework* Laravel 5.3. *Web app* yang diimplementasi dibagi menjadi tiga jenis *dashboard*, yaitu *dashboard* untuk pengelola, *dashboard* untuk mentor, dan *dashboard* untuk *mentee*. Semua *dashboard* ini berfungsi sebagai antarmuka interaksi pengguna dan basis data untuk menjalankan fungsi-fungsi yang dimiliki sistem.

Berikutnya, hasil implementasi sistem yang berbentuk purwarupa mengalami pengujian. Pengujian yang dilakukan berupa pengujian fungsional dan pengujian penerimaan pengguna. Pengujian fungsional dilakukan untuk memeriksa apakah implementasi rancangan sistem telah menjawab kebutuhan fungsional yang didefinisikan. Hasil pengujian fungsional menunjukkan bahwa SI yang dirancang dapat memenuhi

semua kebutuhan fungsional yang didefinisikan.

Pengujian penerimaan pengguna dilakukan untuk mendapatkan pandangan pengguna terhadap kemampuan SI yang dirancang dalam memenuhi

- b. 28 orang dari total 55 orang mentor Bidikmisi sebagai mentor, dan 16 orang dari total 275 orang *mentee* Bidikmisi sebagai *mentee*.

Pengujian dilakukan dengan memberi penjelasan terlebih dahulu cara untuk mengoperasikan SI. Kemudian, subjek diminta untuk mencoba SI yang dirancang. Setelah itu, subjek diminta untuk memberi penilaian terhadap kegunaan dan kemudahan fitur-fitur SI melalui kuesioner.

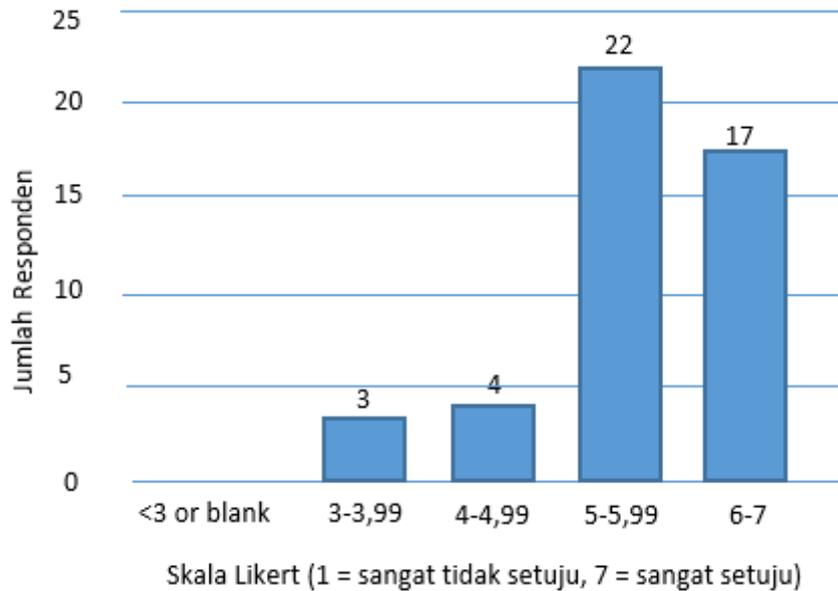
Kegunaan SI yang dirancang dinilai dengan menggunakan kuesioner dari TAM 3. Tujuh butir dari seluruh 25 butir pertanyaan dari TAM 3 untuk penilaian kegunaan suatu teknologi dipilih untuk menjaga agar kuesioner yang digunakan dalam penelitian ini tidak terlalu panjang. Selain itu, tujuannya agar tidak

tujuannya. Pengujian dilakukan secara online kepada 46 responden dengan rincian sebagai berikut.

- a. 2 orang dari total 4 orang staf divisi *mentoring* tim PBM sebagai pengelola *mentoring*,

membingungkan subjek dengan pertanyaan yang mirip dan terkesan berulang, serta relevan dengan keadaan subjek yang baru sekedar mencoba menggunakan SI. Akan tetapi, subjek belum menerapkannya dalam menjalankan kegiatan *mentoring* sehari-hari. Butir-butir pertanyaan yang digunakan dalam kuesioner merupakan pertanyaan berskala Likert mulai dari nilai 1 untuk “sangat tidak setuju” hingga nilai 7 untuk “sangat setuju”.

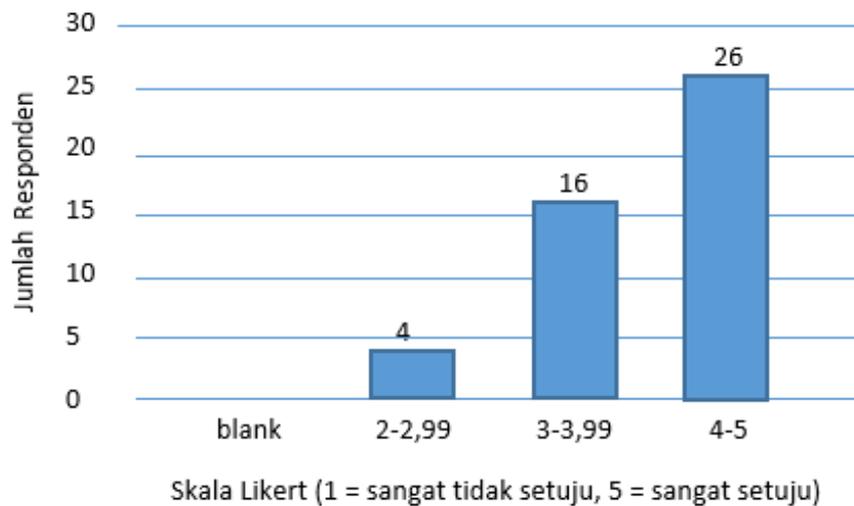
Hasil uji penerimaan pengguna secara umum terhadap kegunaan SI digambarkan di Gambar 5. Berdasarkan hasil penilaian yang dilakukan kepada 46 responden dari berbagai peran tersebut, 85% pengguna menilai bahwa SI berguna bagi pelaksanaan tugas mereka dalam kegiatan *mentoring*.



Grafik 1 Uji Penerimaan Pengguna Secara Umum

Kemudahan SI untuk digunakan dinilai dengan menggunakan kuesioner dari SUPR-Q. Semua butir pertanyaan dari SUPR-Q yang berjumlah 8 buah digunakan dalam kuesioner. Tujuh butir pertanyaan dari SUPR-Q dalam kuesioner tersebut merupakan pertanyaan berskala Likert dengan nilai 1 untuk “sangat tidak setuju” hingga nilai 5 untuk “sangat tidak setuju”. Namun, 1 butir pertanyaan lainnya memiliki skala Likert dengan nilai 1 untuk “sangat tidak setuju” hingga nilai 10 untuk “sangat setuju”. Tingkat kemudahan SI yang dirancang akan direpresentasikan oleh peringkat persentil hasil pengujian.

Hasil uji penilaian pengguna secara umum terhadap kemudahan penggunaan SI digambarkan dalam gambar. Berdasarkan hasil penilaian kemudahan penggunaan tersebut, 57% pengguna menilai SI mudah digunakan dengan rata-rata skor SUPR-Q dari seluruh responden adalah 4,00 dari skala 5. Nilai tersebut menempatkan SI berada pada persentil ke-59 atau dengan kata lain memiliki skor SUPR-Q lebih tinggi dibandingkan 59% *website* lain yang terdapat dalam basis data. Dengan demikian, seluruh pengujian SI kepada pengguna menunjukkan bahwa SI dapat bekerja dengan baik, berguna dalam pengelolaan mentoring, dan mudah digunakan.



Grafik 2 Hasil Uji Penilaian Pengguna Secara Umum Terhadap Kemudahan Penggunaan

SIMPULAN

Pada penelitian ini telah dirancang-bangun SI untuk pengelolaan *mentoring* dengan menerapkan konsep SaaS. Penerapan konsep SaaS dilakukan dengan memenuhi karakteristik utama SaaS, yaitu berarsitektur *single-instance* untuk *multi-tenant*, dikelola secara terpusat dan diakses pengguna lewat internet, serta memiliki skalabilitas dan ketersediaan yang tinggi.

SI untuk pengelolaan *mentoring* menyediakan fitur untuk pengelola *mentoring* serta mentor dan *mentee* yang dikelolanya. Fitur yang terdapat dalam SI terdiri atas pengelolaan anggota, pengelolaan kelompok, pengelolaan pertemuan *mentoring*, publikasi acara, berbagi arsip materi, publikasi formulir evaluasi, dan penerbitan laporan hasil keberjalanan *mentoring*.

Berdasarkan hasil pengujian diperoleh hasil sebagai berikut.

a. Rancangan SI telah memenuhi semua kebutuhan fungsional melalui implementasi purwarupa

web apps dan secara konseptual melalui rancangan.

b. Hasil uji penerimaan pengguna lewat kuesioner dengan 46 responden menunjukkan bahwa rancangan SI ini berguna dalam membantu kegiatan operasional *mentoring* (85%) dan mudah digunakan (57%).

DAFTAR PUSTAKA

- Bhagyashree, A., & Vaishali, B. (May 2012) Article: Data Mining in Cloud Computing. IJCA Proceedings on National Conference on Recent Trends in Computing NCRTC(6).
- Bruce, M., & Bridgeland, J. (2014). *The Mentoring Effect: Young People's Perspectives on the Outcomes and Availability of Mentoring*. Civic Enterprises.
- Chong, F., Carraro, G., & Wolter, R. (2006). *Multi-tenant Data Architecture*. Diakses pada 12 April 2017 dari <https://msdn.microsoft.com/en-us/library/aa479086.aspx>.

- Dimiyathi, A. S. (2010, Oktober 22). *Pembinaan Agama Melalui Pendekatan Dakwah Sistem Langsung (DSL) Untuk Menurunkan Angka Tawuran Pelajar SMA/SMK (Studi Kasus: Pelaksanaan DSL di SMK Provinsi DKI Jakarta)*. Diakses pada 17 Mei 2017 dari <https://sholehdimiyathi.wordpress.com>.
- Hunnebeck, L. (2011). *ITIL Service Design*. The Stationary Office
- Sauro, J. (2015). SUPR-Q: A Comprehensive Measure of the Quality of the Website User Experience. *Journal of Usability Studies*, vol. 10, 68-86.
- (2016). *SUPR-Q Full License*. <https://measuringu.com/product/suprq/>, diakses pada 10 Agustus 2017.
- Venkatesh, V. & Bala, H. (2008). *Technology acceptance model 3 and a Research Agenda on Interventions*. *Decision Sciences Journal Compilation*, vol. 39, 273-315.
- Whitten, Jeffrey L., & Bentley, Lonnie D. (2007). *Systems Analysis & Design Methods*. Irwin: McGraw-Hill.
- Zemin, Z. & Xiaofei (2010). Research and Design of Information System for Basic Education Based on SaaS. 467 - 470. 10.1109/FITME.2009.123.