

Rancang Bangun Aplikasi Sistem Informasi Manajemen Penggunaan Ruang Kuliah (Studi Kasus Jurusan Teknik Elektro dan Komputer Universitas Syiah Kuala)

Munawar Abdullah¹, Ardiansyah², Afdhal³

^{1,2,3}Jurusan Teknik Elektro dan Komputer, Fakultas Teknik, Universitas Syiah Kuala
Jl. Tgk. Syech Abdulrrauf No.7 Darussalam, Banda Aceh 23111, Indonesia

¹munawara682@yahoo.com

²ardiansyah@unsyiah.ac.id

³afdhal@unsyiah.ac.id

Abstrak— Dalam perkuliahan, ruangan merupakan salah satu fasilitas yang dibutuhkan karena perkuliahan dengan tatap muka masih diterapkan di hampir setiap perguruan tinggi di Indonesia. Pengelolaan ruang kuliah di Jurusan Teknik Elektro dan Komputer Universitas Syiah Kuala masih belum sistematis karena informasi yang diberikan hanya berpedoman pada kertas jadwal kuliah yang telah ditetapkan sejak awal semester. Sehingga informasi yang diterima sering tidak sesuai dengan kejadian di lapangan. Dalam pengelolaan ruang yang belum sistematis ini timbul beberapa masalah seperti, dosen tidak dapat mengetahui ruangan mana yang dapat digunakan, ataupun dosen telah melakukan konfirmasi untuk menggunakan ruangan pada jadwal tertentu, tetapi saat ruangan ingin digunakan ternyata ada dosen lain di ruang tersebut. Permasalahan-permasalahan ini tentunya dapat membuat kegiatan perkuliahan tidak bisa berjalan dengan maksimal. Penelitian ini bertujuan untuk merancang dan membangun sebuah sistem informasi manajemen penggunaan ruang kuliah dengan memanfaatkan aplikasi *mobile*, aplikasi *web* dan *QR code*. Aplikasi *mobile* dikembangkan berbasis *hybrid* dan menggunakan *RESTful web service*. Hasil dari penelitian ini adalah terbentuknya aplikasi untuk mengetahui informasi jadwal penggunaan ruang kuliah dan dapat melakukan *booking* ruangan saat pergantian jadwal.

Kata Kunci— aplikasi *mobile*, aplikasi *web*, *QR code*, *hybrid*, *RESTful web service*.

I. PENDAHULUAN

Kegiatan perkuliahan merupakan salah satu kegiatan utama pada setiap perguruan tinggi. Agar kegiatan perkuliahan dapat berlangsung dengan baik, tentu dibutuhkan fasilitas-fasilitas yang akan menunjang terlaksananya kegiatan perkuliahan tersebut. Salah satu fasilitas yang dibutuhkan adalah ruangan kuliah. Hal ini dikarenakan kegiatan perkuliahan dengan sistem tatap muka masih diterapkan di hampir setiap perguruan tinggi di Indonesia. Kegiatan perkuliahan seharusnya dilaksanakan sesuai dengan jadwal dan ruang kuliah yang telah

ditetapkan sejak awal semester. Namun karena berbagai sebab, terkadang ada kegiatan perkuliahan yang tidak dapat dilaksanakan sesuai dengan jadwal yang telah ditetapkan. Hal ini mengakibatkan terjadi pergantian jadwal perkuliahan, sehingga dosen harus menetapkan ulang jadwal dan ruang kuliah yang baru untuk kegiatan perkuliahan.

Dari hasil wawancara dengan beberapa dosen dan pihak administrasi, diketahui bahwa dalam proses pencarian ruang kuliah ganti di Jurusan Teknik Elektro dan Komputer Universitas Syiah Kuala, biasanya dosen menanyakan ruangan yang dapat digunakan untuk kegiatan perkuliahan kepada pihak administrasi. Pihak administrasi memberikan informasi ruangan yang dapat digunakan kepada dosen berdasarkan pada kertas jadwal kuliah yang telah ditetapkan sejak awal semester. Ditemukan kendala bahwa informasi yang diberikan hanya berdasarkan pada asumsi dari jadwal yang telah ditetapkan sejak awal semester, sehingga informasi yang diterima sering tidak sesuai dengan kejadian di lapangan. Sampai saat ini pengelolaan ruang kuliah masih belum sistematis sehingga timbul beberapa masalah seperti, (1) dosen tidak dapat mengetahui ruangan mana yang dapat digunakan, (2) dosen telah melakukan konfirmasi pemakaian ruangan kepada pihak administrasi pada jadwal tertentu, tetapi saat ruangan ingin digunakan ternyata ada dosen lain di ruangan tersebut. Permasalahan-permasalahan ini tentunya dapat membuat kegiatan perkuliahan tidak bisa berjalan dengan maksimal.

Untuk menangani masalah tersebut, dalam penelitian ini akan dirancang sebuah sistem informasi manajemen penggunaan ruang kuliah yang menampilkan informasi jadwal penggunaan ruang dan dapat melakukan *booking* ruangan saat pergantian jadwal. Sistem informasi manajemen penggunaan ruang ini dibangun dengan memanfaatkan aplikasi *mobile*, aplikasi *web*, dan juga *Quick Response code* (*QR code*). Tujuan dari perancangan dan pembuatan sistem ini adalah untuk menyediakan suatu

alternatif solusi yang dapat membantu menyelesaikan permasalahan ruang kuliah seperti dijelaskan di atas.

II. DASAR TEORI

A. Android

Android adalah sebuah sistem operasi berbasis *Linux* yang telah mengalami modifikasi dan diperuntukkan bagi perangkat *mobile*. Sistem operasi ini awalnya dikembangkan oleh *Android Inc.* dengan dukungan finansial dari *Google*, namun kemudian pada tahun 2005 *Google* mengambil alih dengan membelinya [1]. *Android* diresmikan pada tahun 2007 seiring dengan berdirinya *Open Handset Alliance* (OHA), konsorsium dari perusahaan *hardware*, *software*, dan perusahaan telekomunikasi yang ditujukan untuk memajukan standar perangkat seluler. *Smartphone* pertama yang tersedia untuk publik dengan sistem operasi *Android* adalah *HTC Dream*, yang dirilis pada 22 Oktober 2008. Terdapat 2 jenis distributor resmi dari sistem operasi *Android*, yang pertama adalah *Google Mail Services* yang memiliki dukungan penuh *Google* dan yang kedua adalah distributor tanpa dukungan dari *Google* yang sering disebut *Open Handset Distribution* (OHD). *Android* meluncurkan kodenya berdasarkan lisensi *open-source* sehingga dapat dimodifikasi secara bebas dan didistribusikan oleh para pembuat perangkat dan juga para pengembang aplikasi [2].

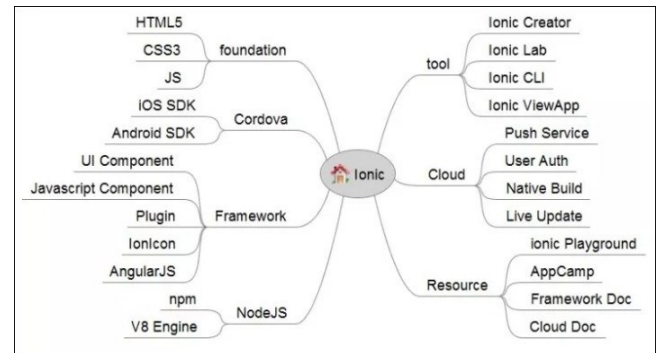
B. Ionic Framework

Ionic Framework adalah sebuah *Software Development Kit* (SDK atau devkit) *open source* yang bisa digunakan pengembang untuk membangun aplikasi *mobile* dengan performansi tinggi menggunakan teknologi *web* seperti *Hypertext Markup Language* (HTML), *Cascading Style Sheet* (CSS), dan *JavaScript* [3]. Diagram Ekosistem *Ionic Framework* dan hubungannya dengan teknologi lain dapat dilihat pada Gambar 1. *Ionic* merupakan *framework* yang dikhususkan untuk membangun aplikasi *mobile* berbasis *hybrid*. Karena berbasis *hybrid*, pengembangan aplikasi yang dilakukan sekali, namun dapat digunakan untuk beberapa sistem operasi. Dengan menggunakan *Ionic*, para pengembang bisa membuat aplikasi lintas *platform* seperti untuk *Android* dan *iOS*. *Ionic Framework* dikembangkan oleh perusahaan *software* Drifty Co. yang didirikan Ben Sperry and Max Lynch pada tahun 2012 [4].

C. Restful Web Service

Web service adalah standar yang digunakan untuk melakukan perpindahan data antar *state*. *Web service* yang berbasis arsitektur REST dikenal sebagai *RESTful web service*. REST diperkenalkan oleh Roy Fielding pada tahun 2000. *Representational State Transfer* (REST) merupakan standar arsitektur komunikasi yang sering diterapkan dalam pengembangan layanan *web*. Umumnya menggunakan protokol *Hypertext Transfer Protocol* (HTTP) untuk komunikasi data. Pada arsitektur REST, REST *server* menyediakan *resource* dan REST *client*

mengakses *resource* tersebut untuk penggunaan selanjutnya. Setiap *resource* diidentifikasi oleh *Universal Resource Identifier* (URI) atau global ID. Pada umumnya format yang digunakan adalah XML dan JSON. *Method* HTTP yang umum digunakan pada *RESTful web service* dapat dilihat pada Tabel I.



Gambar 1 Ionic Framework [5]

TABLE I
METHOD HTTP PADA RESTFUL WEB SERVICE [6]

HTTP method	Operasi yang dilakukan
POST	digunakan untuk membuat <i>resource</i>
GET	digunakan untuk mengambil <i>resource</i>
PUT	digunakan untuk memperbarui <i>resource</i>
DELETE	digunakan untuk menghapus <i>resource</i>

D. Swagger

Swagger adalah sebuah *framework* yang digunakan untuk merancang, membuat dokumentasi dan pengujian *Restful API*. *Swagger* menggunakan bahasa pemrograman *Yet Another Markup Language* (YAML) untuk membuat rancangan *Restful API* [7].

E. JSON

JavaScript Object Notation (JSON) merupakan bagian dari bahasa pemrograman *JavaScript*. Meskipun JSON bagian dari *JavaScript*, JSON tidak termasuk kedalam bahasa pemrograman, melainkan sebagai format yang digunakan untuk pertukaran data. JSON merupakan format data yang universal, sehingga bisa digunakan dalam berbagai bahasa pemrograman. Pada tahun 2001, JSON diadopsi oleh *Asynchronous JavaScript and XML* (AJAX), JSON diperkenalkan sebagai pengganti format *Extensible Markup Language* (XML) karena memiliki kemudahan dalam proses pertukaran data dan konsumsi paket data yang lebih sedikit. JSON hanya bisa mewakili nilai *string*, *number*, *object*, *array*, *true*, *false*, dan *null* [8]. Contoh sintaks JSON dapat dilihat pada Gambar 2.

F. QR Code

Quick Response (QR) *code* adalah suatu jenis kode matriks atau kode batang dua dimensi yang dikembangkan oleh sebuah perusahaan jepang *Denso Wave* dan dipublikasikan pada tahun 1994. Berbeda dengan *barcode*

yang hanya dapat menyimpan informasi secara horizontal, QR code mampu menyimpan informasi secara horizontal dan vertikal. Oleh karena itu, QR code dapat menampung informasi yang lebih banyak daripada barcode [9]. Alat yang digunakan untuk membaca QR code disebut QR code scanner. Umumnya QR code scanner bukanlah alat terpisah, namun tersedia dalam bentuk aplikasi di *smartphone* seperti *Android* atau *iPhone*. Tujuan utama digunakan QR code adalah untuk memudahkan pengguna mengakses informasi. Contoh QR code dapat dilihat pada Gambar 3.

```
{
  "name" : "munawar",
  "age" : 23,
  "social_media" : ["facebook", "twitter", "instagram"],
  "online" : true
}
```

Gambar 2 Contoh sintaks JSON



Gambar 3 Contoh QR code

G. Blackbox Testing

Blackbox testing adalah teknik pengujian yang dilakukan dengan memeriksa *detail* aplikasi seperti tampilan aplikasi, *input-output*, dan fitur-fitur pada aplikasi sehingga dapat berjalan sesuai yang direncanakan. *Blackbox testing* dilakukan tanpa perlu melihat kode program, pengujian hanya mengamati hasil eksekusi dan memeriksa fungsional [10]. *Blackbox testing* dapat mengatasi masalah seperti ketidaksesuaian tampilan, kesalahan dalam struktur data atau akses *database* eksternal, dan fitur-fitur yang tidak benar atau hilang.

H. Usability Testing

Usability testing adalah teknik pengujian sebuah produk yang dilakukan dengan cara mengujinya langsung ke pengguna. Tujuan dari pengujian *usability* adalah untuk mengetahui kualitas dari produk yang dibuat, sehingga dalam pengembangan lebih lanjut dapat meningkatkan kualitas dari produk [11]. Dalam *Usability testing*, tester akan memberikan arahan berupa tugas khusus kepada pengguna, contohnya "berbelanja sepatu merek X dengan

harga Y". Kemudian, pengguna mencoba melakukan tugas tersebut dan tester mengamati gerak-gerik dan perilaku pengguna ketika berinteraksi dengan produk.

Dalam penelitian ini, pengukuran *usability* menggunakan metode *System Usability Scale* (SUS). SUS merupakan paket pengujian *usability* yang efektif digunakan pada berbagai produk [12]. Metode SUS ini tidak memerlukan jumlah responden yang banyak, jumlah responden bisa 2 orang [13], sehingga tidak membutuhkan waktu yang lama. Dalam kuesioner SUS terdapat 10 pernyataan seperti berikut [12]:

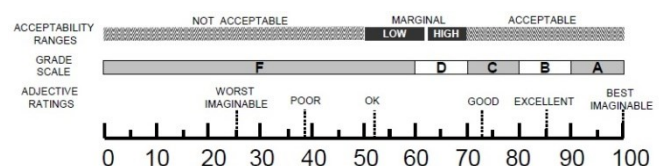
1. Saya rasa saya akan sering menggunakan aplikasi ini.
2. Saya merasa aplikasi terlalu kompleks padahal sebenarnya dapat dibuat sederhana.
3. Saya rasa aplikasi mudah untuk digunakan.
4. Saya rasa saya membutuhkan bantuan dari orang teknis untuk dapat menggunakan aplikasi ini.
5. Saya menemukan bahwa terdapat berbagai macam fungsi yang terintegrasi dengan baik dalam aplikasi.
6. Saya rasa banyak hal yang tidak konsisten terdapat pada aplikasi.
7. Saya rasa mayoritas pengguna akan belajar menggunakan aplikasi ini secara cepat.
8. Saya menemukan bahwa aplikasi sangat tidak praktis.
9. Saya sangat percaya dalam menggunakan aplikasi ini.
10. Saya harus belajar banyak hal terlebih dahulu sebelum saya dapat menggunakan aplikasi ini.

Pada setiap pernyataan di atas terdapat pilihan jawaban dengan skala bernilai dari 1-5.

- STS = Sangat Tidak Setuju, bernilai 1
- TS = Tidak Setuju, bernilai 2
- RG = Ragu-Ragu, bernilai 3
- S = Setuju, bernilai 4
- SS = Sangat Setuju, bernilai 5

Perhitungan nilai skor untuk setiap pernyataan hasil kuesioner SUS dengan aturan sebagai berikut:

- Untuk pernyataan nomor ganjil (1,3,5,7,9), nilai skala setiap jawaban yang dipilih, dikurangi 1.
- Untuk pernyataan nomor genap (2,4,6,8,10), 5 dikurangi nilai skala setiap jawaban yang dipilih.



Gambar 4 Konversi skala nilai SUS [12]

Setiap pernyataan akan mendapat skor perhitungan berkisar dari 0-4. Selanjutnya nilai skor untuk setiap

pernyataan dikalikan dengan 2.5 dan kemudian dijumlahkan. Jumlah skor SUS yang dihasilkan masing-masing responden akan berkisar dari 0 hingga 100. Untuk mendapatkan nilai total SUS dari seluruh pengujian adalah dengan menjumlahkan skor SUS seluruh responden dan kemudian dibagi dengan jumlah responden. Selanjutnya skor SUS dapat dikonversikan dengan skala seperti pada Gambar 4.

III. METODE PENELITIAN

A. Tahapan Penelitian

Tahapan penelitian yang dilakukan pada penelitian ini diperlihatkan pada Gambar 5. Tahapan awal yang dilakukan yaitu mencari referensi-referensi yang berkaitan, baik dari buku, artikel, jurnal, dan lain sebagainya untuk bahan studi yang berkaitan dengan penelitian ini.

Kemudian dilanjutkan dengan analisa kebutuhan, pada tahapan analisa kebutuhan dilakukan pengumpulan data yang berupa permasalahan-permasalahan yang terjadi saat ini pada pencarian ruang kuliah dan *booking* ruangan untuk kuliah ganti. Pengumpulan data dilakukan dengan mewawancarai dosen dan pihak administrasi.

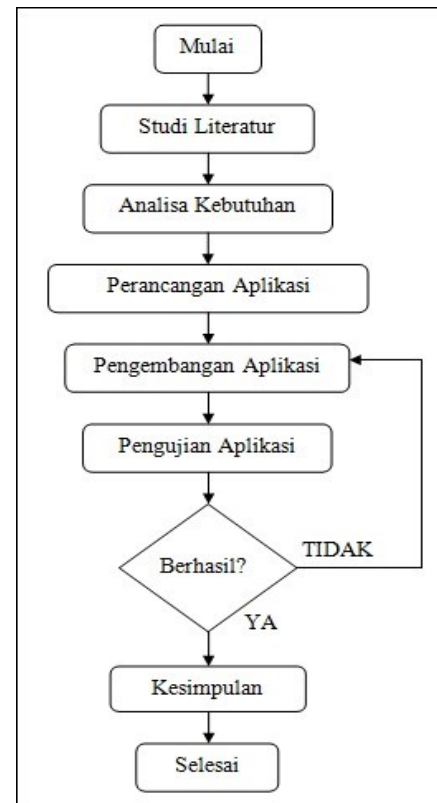
Pada tahapan selanjutnya dilakukan perancangan aplikasi dan layanan *web*. Perancangan API pada layanan *web* dilakukan menggunakan *Swagger Framework*. Pada tahapan ini akan menghasilkan rancangan aplikasi dan rancangan *Restful API Web Service*.

Berdasarkan rancangan yang sudah dibuat, selanjutnya dilakukan pengembangan aplikasi dan layanan *web*. Setelah dilakukan pengembangan aplikasi, kemudian pada tahapan akhir dilakukan pengujian dan kesimpulan dari penelitian. Adapun metode pengujian yang dilakukan pada tahap pengujian aplikasi yaitu *BlackBox Testing* dan *Usability Testing*.

IV. HASIL DAN PEMBAHASAN

A. Analisa Kebutuhan

Pada sistem sekarang, saat memerlukan sebuah ruangan untuk kuliah pengganti, dosen harus datang ke pihak administrasi untuk menanyakan ruangan yang bisa digunakan. Berdasarkan wawancara yang telah dilakukan dengan pihak administrasi dan dosen, didapatkan data yang menunjukkan bahwa sistem pengelolaan ruangan kuliah saat ini masih belum sistematis karena hanya berpedoman pada kertas jadwal kuliah yang ditetapkan sejak awal semester, sehingga informasi yang diberikan sering tidak sesuai dengan kejadian di lapangan. Untuk menangani masalah tersebut, dalam penelitian ini dibangun aplikasi sistem informasi manajemen penggunaan ruang kuliah sebagai sebuah alternatif solusi yang dapat membantu menyelesaikan permasalahan di atas. Aplikasi yang dibangun adalah aplikasi *web* dan aplikasi *mobile* untuk kelompok pengguna yang berbeda. Ada 2 kelompok pengguna yang berinteraksi dalam sistem, yaitu Admin dan Dosen.



Gambar 5 Diagram alir tahapan penelitian



Gambar 6 Persona admin

1) *Admin*: Pengguna aplikasi *web* yang bertugas mengelola sistem informasi manajemen penggunaan ruang kuliah. Admin mendaftarkan dosen, melakukan penginputan data jadwal kuliah di awal semester, dan mengelola informasi fasilitas ruangan. Persona dari pengguna admin dapat dilihat pada Gambar 6.

2) *Dosen*: Pengguna aplikasi *mobile* untuk melihat informasi ruangan yang dapat digunakan, *booking* ruangan saat pergantian jadwal, melapor fasilitas ruangan, dan melakukan absensi menggunakan *QR code*. Persona dari pengguna dosen dapat dilihat pada Gambar 7. Berdasarkan user story dari masing-masing pengguna di atas, Gambar 8 merupakan *use case diagram* yang menggambarkan interaksi pengguna dengan sistem. Interaksi admin dengan aplikasi *web* adalah mendaftarkan dosen ke dalam sistem

dan melakukan penginputan data jadwal kuliah pada awal semester. Data jadwal kuliah berisi informasi matakuliah dan ruangan yang juga harus dimasukkan oleh admin sebelum memasukkan atau membuat jadwal kuliah. Selain itu, admin juga mengelola informasi fasilitas ruangan, baik memperbarui informasi fasilitas ruangan sehingga dosen dapat mengetahui informasi fasilitas sebuah ruangan, atau membaca informasi laporan fasilitas ruangan yang dikirim oleh dosen melalui aplikasi mobile. Interaksi dosen dengan aplikasi mobile adalah melakukan booking ruangan saat pergantian jadwal kuliah dan melakukan absensi masuk menggunakan QR code. Melalui aplikasi mobile, dosen juga dapat melaporkan fasilitas sebuah ruangan dengan cara mengirim laporan pada setiap ruangan tertentu. Informasi laporan yang dikirimkan akan masuk ke aplikasi web yang digunakan oleh admin.

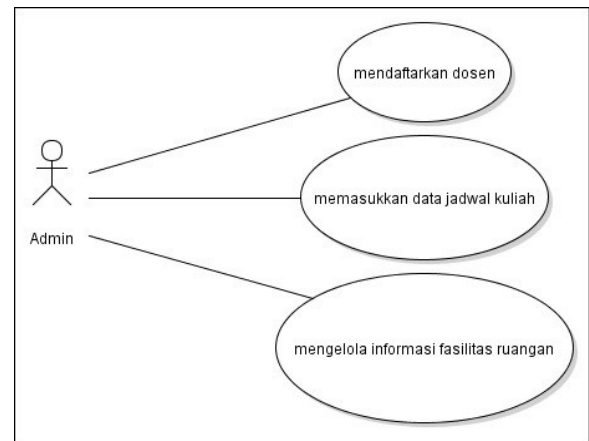
B. Perancangan Sistem

1) *Prinsip Kerja dan Arsitektur Sistem*: Prinsip kerja sistem menggambarkan bagaimana sistem bekerja, prinsip kerja sistem dapat dilihat pada Gambar 10. Data yang telah tersimpan di dalam database dapat digunakan oleh aplikasi mobile menggunakan layanan Restful API Web Service dengan method HTTP dan format data JSON. Method HTTP yang digunakan adalah method GET, POST, dan PUT. Method GET digunakan untuk mengambil atau membaca data yang tersimpan didalam database melalui web service. Sedangkan method POST dan PUT digunakan oleh aplikasi mobile untuk memperbaharui database melalui web service. Aplikasi web melakukan komunikasi data secara langsung ke database seperti perintah create, read, update, dan delete.

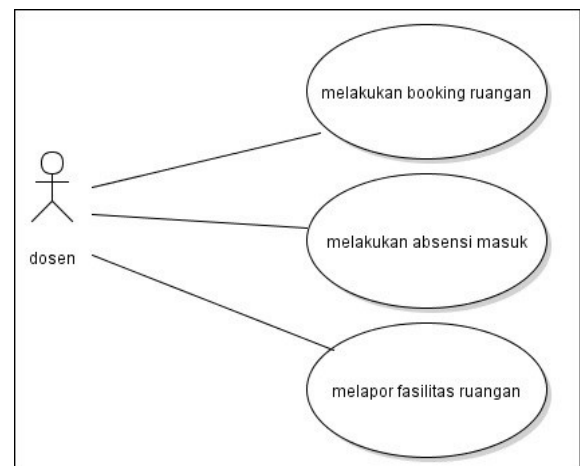
2) *Entity Relationship Diagram*: Gambar 11 menggambarkan entitas apa saja yang berada pada *backend* sistem informasi manajemen penggunaan ruang kuliah dan bagaimana hubungan dari setiap entitas dengan entitas lain.



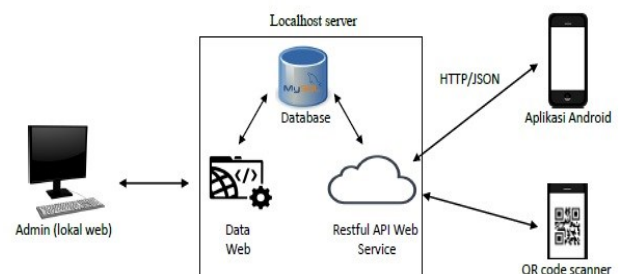
Gambar 7 Persona dosen



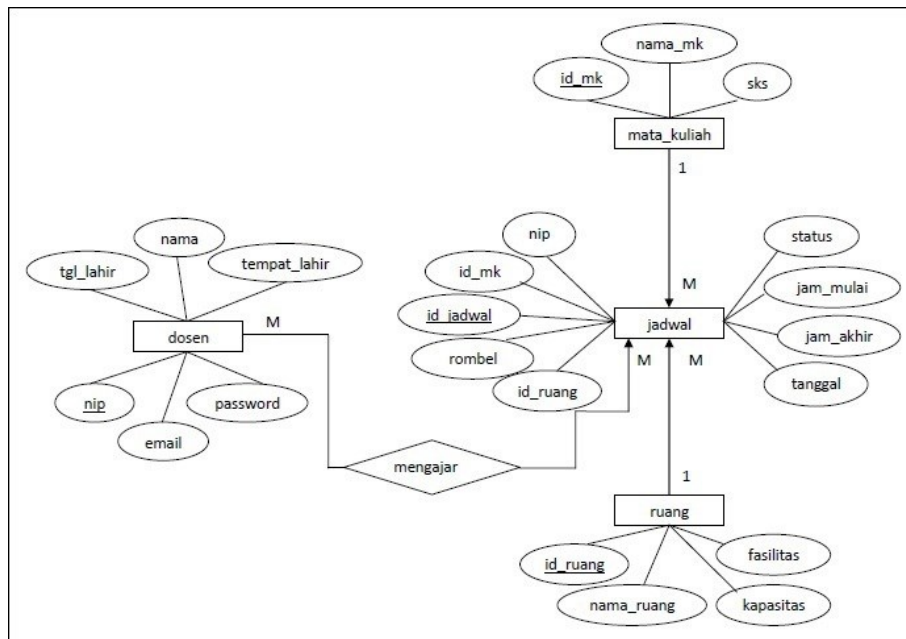
Gambar 8 Use case diagram admin



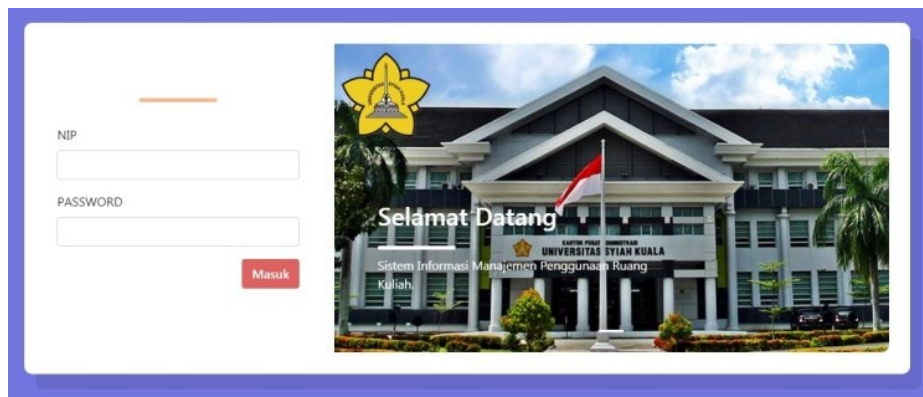
Gambar 9 Use case diagram dosen



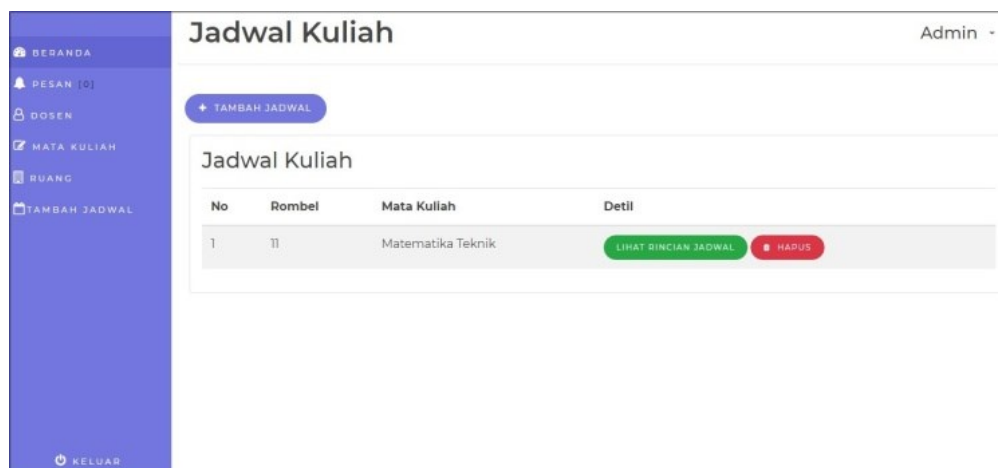
Gambar 10 Diagram prinsip kerja sistem



Gambar 11 Entity Relationship Diagram



Gambar 12 Tampilan login aplikasi web



Gambar 13 Tampilan menu pada aplikasi web

3) *Perancangan Restful Web Service*: Perancangan API bertujuan untuk mengidentifikasi apa saja kebutuhan API sesuai dengan spesifikasi yang akan dibangun. Dengan dilakukan perancangan API terlebih dahulu, maka akan lebih memudahkan pengembang dalam membangun *Restful Web Service*. Perancangan dilakukan menggunakan *Swagger Framework*.

C. Pengujian dan Implementasi

Aplikasi *mobile* melakukan komunikasi data dengan *backend* melalui *Restful Web Service* yang berada di dalam *server*. *Web Service* berisi API atau fungsi-fungsi yang menyediakan *resource* untuk digunakan. Sebelum digunakan, API atau fungsi pada *Restful Web Service* yang telah dibuat dapat diuji coba menggunakan *software Postman*.

Sistem informasi manajemen penggunaan ruang kuliah dibangun dengan mengembangkan 2 (dua) aplikasi, yaitu aplikasi *web* dan aplikasi *mobile*. Aplikasi *web* dikembangkan menggunakan *Framework CodeIgniter* dengan metode *Model View Controller (MVC)*.

Aplikasi *web* digunakan oleh admin untuk mengelola sistem informasi manajemen penggunaan ruang kuliah. Admin yang menggunakan aplikasi *web* melakukan login melalui *form* pada Gambar 12.

Admin memasukkan NIP dan *password* sebagai informasi untuk masuk ke aplikasi. Selanjutnya admin akan diarahkan ke halaman beranda seperti pada Gambar 13. Pada aplikasi *web* terdapat 6 (enam) menu sesuai dengan kegunaan masing-masing. Pada menu "beranda" berisi daftar jadwal kuliah yang telah dimasukkan oleh admin ke dalam aplikasi. Menu "pesan" berisi informasi mengenai laporan-laporan fasilitas pada ruangan kuliah yang dilaporkan oleh dosen. Menu "dosen" berfungsi untuk mendaftarkan dosen. Menu "mata kuliah" dan "ruang" berfungsi untuk memasukkan data matakuliah dan ruang kuliah. Terakhir menu "tambah jadwal" berfungsi untuk memasukkan data jadwal kuliah pada awal semester.

Untuk menggunakan aplikasi *mobile* dosen juga harus melakukan *login*. Dosen melakukan *login* ke aplikasi *mobile* dengan memasukkan NIP dan *password* yang sudah terdaftar sebelumnya. Halaman *login* seperti pada Gambar 14.

Pengguna yang berhasil login akan diarahkan ke menu "jadwal". Tampilan halaman menu "jadwal" seperti pada Gambar 15. Pada aplikasi *mobile* terdapat 3 (tiga) menu. Pada menu "jadwal" terdapat daftar mata kuliah yang diasuh oleh dosen yang bersangkutan. Pada menu "ruang" berisi daftar nama-nama ruang yang ada dalam sistem. Terakhir menu "profil" yang berisi informasi nip dan nama pengguna yang sedang login, dan terdapat tombol "keluar".

Aplikasi yang telah dikembangkan selanjutnya dilakukan pengujian menggunakan *Blackbox Testing* dan *Usability Testing*. Pengujian *Blackbox Testing* dilakukan untuk memverifikasi dan validasi kesesuaian fungsi dari program aplikasi yang telah dibuat. Pengujian dilakukan

dengan cara memberikan sejumlah *input* pada program sesuai dengan skenario pengujian. Dan melihat apakah program dapat menghasilkan *output* yang sesuai dengan kebutuhan fungsionalnya. Dari seluruh skenario pengujian *Blackbox Testing* yang telah dilakukan pada aplikasi *web* dan *mobile*, semuanya berjalan sesuai dengan hasil yang diharapkan atau *valid*.

Gambar 14 Tampilan login aplikasi mobile

Gambar 15 Tampilan menu pada aplikasi mobile

Selanjutnya dilakukan pengujian *Usability Testing* terhadap 3 pengguna aplikasi *web* dan 10 pengguna aplikasi *mobile*. Pada tahap pengujian *usability*, para responden yang terpilih akan melakukan tugas-tugas sesuai skenario pengujian. Pengukuran *usability* dilakukan menggunakan metode SUS dengan melakukan pengujian aplikasi secara langsung kepada pengguna dan kemudian pengguna diminta untuk mengisi kuesioner berdasarkan pengalamannya menggunakan aplikasi. Hasil yang diharapkan dari pengujian *usability* adalah untuk mengetahui kualitas atau tingkat kemudahan penggunaan aplikasi, sehingga setelah dilakukan pengujian dalam pengembangan lebih lanjut dapat memperbaiki kesalahan-kesalahan yang ditemukan.

Pengguna akan menggunakan aplikasi setelah diberikan skenario pengujian. Pada saat pengujian dilakukan, responden akan diamati gerak-gerik dan perilaku ketika berinteraksi dengan aplikasi. Setiap skenario pengujian, pengguna memiliki 1 kesempatan untuk menggunakan aplikasi untuk melakukan tugas sesuai skenario yang diberikan. Berdasarkan hasil pengamatan yang telah dilakukan saat pengujian aplikasi *web* dan aplikasi *mobile*, tidak terdapat kesalahan yang dominan dilakukan oleh seluruh pengguna. Semua pengguna berhasil menggunakan aplikasi sesuai dengan skenario yang diharapkan. Kecuali pada pengujian aplikasi *web* saat mengunduh absensi, semua pengguna tidak berhasil karena tempat pengunduhan absensi sulit ditemukan. Hal ini menunjukkan bahwa fitur mengunduh absensi perlu diperbaiki.

Berdasarkan pengujian *Usability Testing* dan pengolahan data yang telah dilakukan didapatkan nilai SUS aplikasi *web* sebesar 82.5 dan nilai SUS aplikasi *mobile* sebesar 81. Kedua nilai SUS yang didapatkan tersebut bernilai B jika dikonversikan menurut standar A. Bangor, P. Kortum, dan J. Miller [12], masuk dalam kategori *Acceptable* jika dikonversikan menurut rating penerimaan, dan dikategorikan ke dalam *Excellent* berdasarkan *adjective rating* seperti pada Gambar 4.

REFERENSI

- [1] P. Kaur and S. Sharma, "Google Android a mobile platform: A review," *2014 Recent Advances in Engineering and Computational Sciences (RAECS)*, Chandigarh, 2014, pp. 1-5.
- [2] A. Ayyasamy, "Survey on Android application advancement and security," *2015 Seventh International Conference on Advanced Computing (ICoAC)*, Chennai, 2015, pp. 1-4.
- [3] "Concepts Ionic Framework." [Online]. Available: <https://ionicframework.com/docs/intro/concepts/>. [Accessed: 05-August-2018].
- [4] "All about Ionic." [Online]. Available: <https://ionicframework.com/about>. [Accessed: 05-August-2018].
- [5] "Mengenal Ionic Framework." [Online]. Available: <http://www.candra.web.id/mengenal-ionic-framework/>. [Accessed: 15-August-2018].
- [6]
- [7] K. Murthy, P. S. Ashok kumar, P. Palagi and N. Bhasaha, "Implication of Restful Web Services in Bigdata Analytics," *2017 2nd International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS)*, Bengaluru, India, 2017, pp. 1-5.
- [8] "Swagger." [Online]. Available: <https://swagger.io/>. [Accessed: 20-Jan-2019].
- [9] "Introducing JSON." [Online]. Available: <https://www.json.org/>. [Accessed: 10-August-2018].
- [10] S. Tiwari, "An Introduction to QR Code Technology," *2016 International Conference on Information Technology (ICIT)*, Bhubaneswar, 2016, pp. 39-44.
- [11] Gerald D. Everett; Raymond McLeod, "Functional Testing," in *Software Testing: Testing Across the Entire Software Development Life Cycle*, IEEE, 2007, pp.
- [12] J. Rubin, D. Chisnell, *Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests*, 2nd Ed. Indianapolis, Indiana: John Wiley & Sons, Inc, 2008.
- [13] A. Bangor, P. Kortum, and J. Miller, "Determining what individual SUS scores mean: Adding an adjective rating scale," *Journal of Usability Studies*, vol. 4, no. 3, pp. 114-123, 2009.
- [14] "Measuring Usability With The System Usability Scale (SUS)." [Online]. Available: <https://measuringu.com/sus/>. [Accessed: 20-Jan-2019].

V. KESIMPULAN

Berdasarkan hasil penelitian, kesimpulan dari penelitian ini adalah: aplikasi *web* dan *mobile* yang telah dikembangkan dapat digunakan oleh dosen dan admin untuk sistem informasi manajemen penggunaan ruang kuliah. Dan skor SUS yang didapatkan aplikasi *web* dan aplikasi *mobile* bernilai B atau masuk dalam kategori *Acceptable* jika dikonversikan menurut *rating* penerimaan, dan dikategorikan ke dalam *Excellent* berdasarkan *adjective rating*.

UCAPAN TERIMA KASIH

Dengan selesainya Karya Ilmiah ini, penulis mengucapkan terima kasih kepada orang tua dan keluarga yang telah banyak memberikan bantuan, doa, semangat dan dukungan selama ini. Bapak Ardiansyah, BSEE., M.Sc. dan Bapak Afdhal, S.T., M.Sc. selaku Dosen Pembimbing I, Dosen Pembimbing II yang telah menyediakan banyak waktu, tenaga, dan pikiran untuk mengarahkan saya dalam penyusunan Karya Ilmiah ini. Bapak Rahmad Dawood, S.Kom., M.Sc, Bapak Kahlil Muchtar, S.T., M.Eng., Ph.D, dan Ibu Maya Fitria, S.Kom., M.Sc. selaku Ketua Sidang, Dosen Penguji I dan Dosen Penguji II. Bapak Dr. Nasaruddin, ST., M.Eng selaku Ketua Jurusan Teknik Elektro dan Komputer Fakultas Teknik Universitas Syiah Kuala. Bapak Elizar, S.T., M.Sc., M.Phil. selaku Sekretaris Jurusan Teknik Elektro dan Komputer Fakultas Teknik Universitas Syiah Kuala. Bapak Zulhelmi, S.T., M.Sc. selaku Ketua Program Studi Teknik Elektro Fakultas Teknik Universitas Syiah Kuala. Seluruh dosen dan staf Jurusan Teknik Elektro dan Komputer Fakultas Teknik Universitas Syiah Kuala. Teman-teman mahasiswa Jurusan Teknik Elektro dan Komputer Fakultas Teknik Universitas Syiah Kuala dan seluruh pihak yang telah ikut membantu dalam penyusunan karya ilmiah ini.