

PENGHASIL KUIS OTOMATIS DENGAN KONSEP LINKED DATA



Disusun Oleh:

N a m a : Reza Nouryan Rachman

NIM : 16523092

**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2020

HALAMAN PENGESAHAN DOSEN PEMBIMBING

PENGHASIL KUIS OTOMATIS DENGAN KONSEP LINKED
DATA

TUGAS AKHIR



Disusun Oleh:

N a m a : Reza Nouryan Rachman

NIM : 16523092

الجامعة الإسلامية
Yogyakarta, 11 Juli 2020

Pembimbing,

A handwritten signature in blue ink, appearing to read 'Hendrik'.

(Hendrik, S.T., M.Eng.)

HALAMAN PENGESAHAN DOSEN PENGUJI

**PENGHASIL KUIS OTOMATIS DENGAN KONSEP LINKED
DATA**

TUGAS AKHIR

Telah dipertahankan di depan sidang pengujian sebagai salah satu syarat untuk
memperoleh gelar Sarjana Komputer dari Program Studi Informatika
di Fakultas Teknologi Industri Universitas Islam Indonesia
Yogyakarta, 2020

Tim Penguji

Hendrik, S.T., M.Eng.

Anggota 1

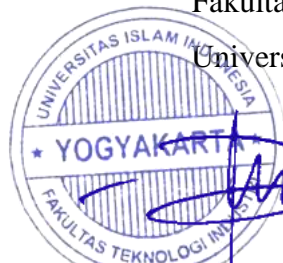
Hanson Prihantoro Putro, S.T., M.T.

Anggota 2

Lizda Iswari, S.T., M.Sc.


 Mengetahui,

Ketua Program Studi Informatika – Program Sarjana
Fakultas Teknologi Industri
Universitas Islam Indonesia




 (Dr. Raden Teduh Dirgahayu, S.T., M.Sc.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Reza Nouryan Rachman

NIM : 16523092

Tugas akhir dengan judul:

PENGHASIL KUIS OTOMATIS DENGAN KONSEP LINKED DATA

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila dikemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung resiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 2020, 20 Juli 2020



(Reza Nouryan Rachman)

HALAMAN PERSEMBAHAN

Alhamdulillah rabbil'alaamiin

Puji dan syukur dipanjatkan ke hadirat Allah SWT karena atas limpahan rahmat dan karunia-Nya sehingga skripsi ini dapat selesai

Terima kasih kepada Orang Tua
(Udin Mapiase dan Masjannah)
atas segala dukungan dan doa yang diberikan.

Terima kasih kepada keluarga besar saya yang sudah memberikan dukungan motivasi yang sangat berarti dan membangun.

Terima kasih kepada Bapak Hendrik, ST., M.Eng.
Terima kasih atas ilmu dan bimbingan yang diberikan.

Terima kasih kepada kerabat, sahabat, dan teman-teman saya yang selalu membantu dan hadir menemani hari-hari saya selama kuliah .

Jazakumullah Khairan Katsiran



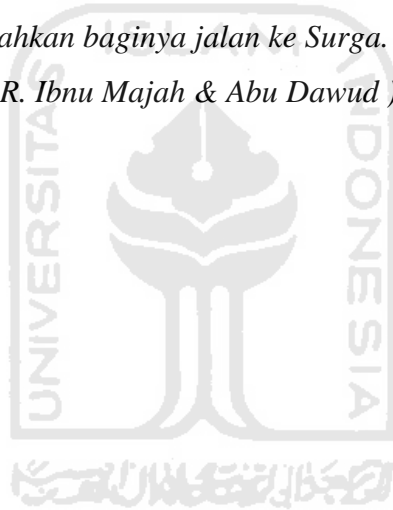
HALAMAN MOTO

Allah tidak membebani seseorang melainkan sesuai dengan kadar kesanggupannya.

(Q.S Al-Baqarah: 286)

Barangsiapa yang menapaki suatu jalan dalam rangka mencari ilmu maka Allah akan memudahkan baginya jalan ke Surga.

(H.R. Ibnu Majah & Abu Dawud)



KATA PENGANTAR

Assalamulaikum Warahmatullahi Wabarakatuh

Alhamdulillah rabbi'l'alamin, penulis panjatkan ke hadirat Allah SWT atas rahmat dan hidayat-NYA dapat menyelesaikan penyusunan laporan tugas akhir dengan baik. Shalawat seta salam tak lupa selalu diucapkan kepada Rasulullah SAW beserta para sahabat, keluarga, dan pengikutnya yang diharapkan syafaatnya di akhir zaman nanti.

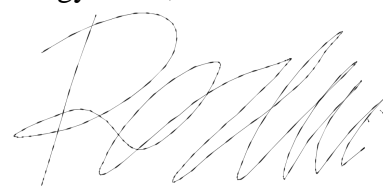
Bimbingan serta bantuan baik secara moril ataupun materiil yang senantiasa datang kepada penulis baik secara langsung maupun tidak langsung selama penyusunan laporan Tugas akhir ini. Maka dari itu dalam kesempatan ini penulis ingin mengucapkan terima kasih kepada:

1. Bapak Fathul Wahid, S.T., M.Sc., Ph.D., selaku Rektor Universitas Islam Indonesia.
2. Bapak Prof. Dr. Ir. Hari Purnomo selaku Dekan Fakultas Teknik Industri Universitas Islam Indonesia.
3. Bapak Hendrik, ST., M.Sc., selaku Ketua Jurusan Informatika Universitas Islam Indonesia dan dosen pembimbing tugas akhir atas bimbingan dan ilmu yang diberikan serta kasabaran beliau selama penyusunan tugas akhir ini.
4. Bapak Raden Teduh Dirgahayu, S.T., M.Sc., selaku ketua Program Studi Informatika Universitas Islam Indonesia.
5. Kedua orang tua, saudara, serta seluruh keluarga yang senantiasa memberikan dukungan dan doa yang tiada hentinya sehingga diberikan kelancaran dalam menyelesaikan tugas akhir ini.

Akhir kata, semoga laporan tugas akhir ini dapat bermanfaat bagi penulis pada khususnya dan pembaca pada umumnya. Dengan kerendahan hati, penulis mengharapkan kritik dan saran yang membangun demi perbaikan penyusunan tugas akhir selanjutnya.

Wassalamu'alaikum Warahmatullahi Wabarakatuh.

Yogyakarta, 11 Juli 2020



(Reza Nouryan Rachman)

SARI

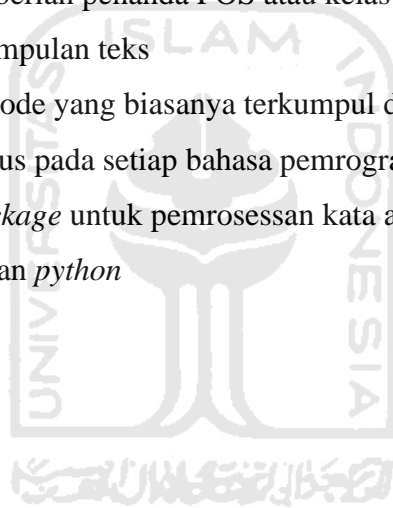
Kuis merupakan cara alternatif yang dapat digunakan untuk mengetahui pemahaman konsep mahasiswa secara interaktif sebagai bentuk timbal balik setelah materi dipaparkan oleh dosen. Kegiatan yang sangat menyenangkan bagi mahasiswa yang biasa digunakan dosen dengan memberikan pertanyaan singkat pada saat kegiatan perkuliahan akan dimulai. Seiring majunya perkembangan teknologi informasi juga membawa perubahan dalam pemanfaatan kuis yang bisa diberikan secara daring sebagai tugas pengganti jika kuliah tatap muka ditunda. Aplikasi kuis online bisa menjadi cara efektif dalam menyajikan kuis kepada mahasiswa namun, diamati bahwa aplikasi penghasil kuis yang ada masih mengalami beberapa keterbatasan umum yaitu kuis yang didapatkan harus dibuat secara manual. Pada penelitian ini penulis akan memaparkan bagaimana metode *Linked Data* dan *Dbpedia* sebagai sumber materi pada aplikasi penghasil kuis otomatis berbasis web dengan *framework* flask yang kerangka metadatanya dimodelkan dalam bentuk *Resource Description Framework* (RDF) dengan melakukan query pada *SPARQL Endpoint*. Metodologi yang dilakukan meliputi tahap analisis kebutuhan, studi literatur, perancangan sistem, dan pada implementasi sistem yang telah jadi akan diuji secara *Black Box Testing* yaitu dengan mengecek fungsionalitas sistem untuk mengetahui kekurangan dan kelebihan sistem. Hasil akhir dari penelitian adalah sistem yang dapat menghasilkan kuis secara otomatis dengan metode *Linked Data* dengan sumber data *Dbpedia* berbasis *web*.

Kata kunci: *linked data*, *SPARQL*, penghasil kuis otomatis, *Dbpedia*

GLOSARIUM

- Compile* proses untuk mengubah berkas kode program dengan berkas lain yang terkait menjadi berkas yang siap untuk dieksekusi oleh sistem operasi secara langsung.
- Bug* kesalahan kode program.
- Black Box Testing* suatu pengujian yang dilakukan oleh pengguna sistem untuk mengecek fungsionalitas sistem yang dibuat apakah sesuai dengan kebutuhan.
- Use Case Diagram* salah satu bentuk diagram UML yang bermanfaat untuk memetakan kebutuhan fungsional dan hubungan interaksi aktor yang bersangkutan dengan suatu sistem.
- Activity Diagram* salah satu bentuk diagram UML yang bermanfaat untuk memetakan urutan aktivitas aktor yang bersangkutan dengan suatu sistem.
- Platform* merupakan gabungan dari kerangka kerja perangkat lunak dan perangkat keras untuk menjalankan sebuah aplikasi atau program.
- Web Application* perangkat lunak yang diakses dengan mesin penjelajah web melalui jaringan internet dan dapat dikembangkan dengan berbagai bahasa pemrograman yang mendukung seperti HTML, Javascript, PHP, Python, CSS, dan SQL.
- Web Framework* sebuah kerangka kerja perangkat lunak yang membantu dalam pengembangan situs web secara mudah dan cepat. Didalamnya tersedia berbagai fungsi siap pakai dalam mengelola akses ke database, pengelolaan antarmuka, dan pengelolaan sesi akses masuk ke dalam web.
- Mockup Aplikasi* sketsa gambaran awal desain antarmuka sistem atau aplikasi yang akan dibuat.
- User Acceptance Test* pengujian yang dilakukan oleh pengguna sistem dengan hasil dokumen hasil pengujian yang sebagai bukti bahwa sistem telah memenuhi kebutuhan serta mengukur langsung kemudahan yang dirasakan pengguna sistem.

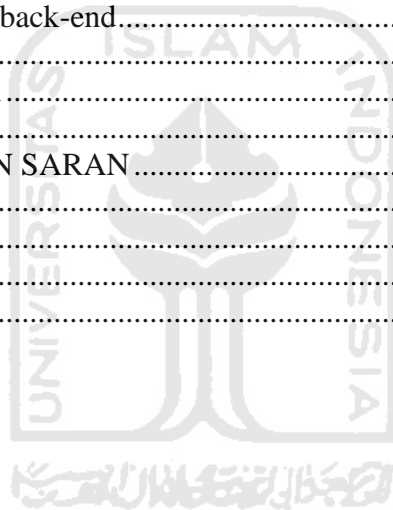
<i>Back-end</i>	bagian latar belakang suatu sistem yang berisi baris kode pemrograman yang berfungsi agar aplikasi dapat berjalan dengan baik dan dapat diakses oleh pengguna.
<i>SPARQL Endpoint</i>	jaringan HTTP yang mampu menerima dan memproses permintaan protokol <i>query</i> SPARQL.
<i>Linked Data</i>	metode untuk mempublikasikan data terstruktur menggunakan kosakata yang dapat dihubungkan bersama dan diinterpretasikan oleh internet.
<i>Python</i>	bahasa pemrograman interpretatif multiguna.
<i>Flask</i>	<i>Web framework</i> khusus untuk pemrograman <i>python</i>
<i>POS tagging</i>	proses pemberian penanda POS atau kelas sintaktik pada tiap kata di dalam sekumpulan teks
<i>Library package</i>	kumpulan code yang biasanya terkumpul dalam sebuah modul dengan fungsi khusus pada setiap bahasa pemrograman.
<i>NLTK</i>	<i>Library package</i> untuk pemrosesan kata atau teks pada bahasa pemrograman <i>python</i>



DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING	ii
HALAMAN PENGESAHAN DOSEN PENGUJI	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR.....	iv
HALAMAN PERSEMBAHAN	v
HALAMAN MOTO	vi
KATA PENGANTAR.....	vii
SARI.....	viii
GLOSARIUM	ix
DAFTAR ISI	xi
DAFTAR TABEL	xiii
DAFTAR GAMBAR.....	xiv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	2
1.5 Manfaat Penelitian	2
1.6 Metodologi Penelitian	3
1.7 Sistematika Penulisan	3
BAB II LANDASAN TEORI.....	5
2.1 SISTEM INFORMASI	5
2.2 WEB SEMANTIK	5
2.3 LINKED DATA.....	5
2.4 DBPEDIA	6
2.5 RDF.....	7
2.6 SPARQL	8
2.7 FLASK.....	9
2.8 NLTK.....	10
2.9 PENELITIAN TERKAIT	10
BAB III METODOLOGI PENELITIAN	12
3.1 Lokasi dan Waktu penelitian.....	12
3.2 Alat Penelitian.....	12
3.3 Metode Penelitian	13
3.4 Studi Literatur	13
3.5 Idenfikasi masalah.....	14
3.6 Analisis Kebutuhan	14
3.6.1 Kebutuhan fungsional.....	14
3.6.2 Kebutuhan non-fungsional	15
3.7 Rancangan Aplikasi	15
3.7.1 <i>Use Case Diagram</i>	15
3.7.2 <i>Activity Diagram</i>	17
3.7.3 <i>Mockup</i> sistem	18
3.7.4 Arsitektur back-end sistem.....	24
3.8 Rancangan Pengujian	25
3.8.1 Pengujian <i>Black Box Testing</i>	25

	xii
3.8.2 Pengujian User Acceptance Test.....	27
3.8.3 Alat Penguji.....	28
3.8.4 Lokasi	28
BAB IV PEMBAHASAN	29
4.1 Batasan Implementasi	29
4.2 Spesifikasi Sistem	29
4.3 Implementasi Sistem	31
4.3.1 Halaman <i>login</i>	31
4.3.2 Halaman <i>sign up</i>	31
4.3.3 Halaman <i>home dashboard</i> mahasiswa	32
4.3.4 Halaman histori nilai kuis mahasiswa	33
4.3.5 Halaman masuk kuis	33
4.3.6 Halaman kuis	34
4.3.7 Halaman <i>home dashboard</i> dosen	34
4.3.8 Halaman tambah kuis	35
4.3.9 Halaman manajemen kuis	35
4.3.10 Halaman <i>user management</i> dosen.....	36
4.3.11 Halaman <i>leaderboard</i>	37
4.3.12 Implementasi back-end.....	37
4.4 Pengujian Sistem.....	44
4.5 Hasil Pengujian sistem.....	48
4.6 Keterbatasan Sistem	48
BAB V KESIMPULAN DAN SARAN	50
5.1 Kesimpulan	50
5.2 Saran.....	50
DAFTAR PUSTAKA.....	52
LAMPIRAN	54



DAFTAR TABEL

Tabel 2. 1 Penelitian Terkait.....	10
Tabel 3. 1 Daftar peralatan penelitian.....	12
Tabel 3. 2 Daftar prioritas kebutuhan fungsional	14
Tabel 3. 3 Kebutuhan non-fungsional perangkat lunak sistem.....	15
Tabel 3. 4 Skenario <i>Black Box Testing</i>	25
Tabel 4. 1 Spesifikasi Sistem	29
Tabel 4. 2 Hasil pengujian <i>Black Box</i>	45
Tabel 4. 3 Hasil pengujian UAT	47
Tabel 4. 4 Persentase hasil pengujian UAT	47



DAFTAR GAMBAR

Gambar 2. 1 tampilan DBpedia.org	7
Gambar 2. 2 hasil tabel format HTML pencarian query dengan SPARQL.....	9
Gambar 3. 1 <i>Flowchart</i> tahapan metode penelitian	13
Gambar 3. 2 <i>Use Case Diagram</i> Pengembangan Aplikasi Penghasil Kuis Otomatis	16
Gambar 3. 3 <i>Activity Diagram</i> Pengembangan Aplikasi Penghasil Kuis Otomatis	17
Gambar 3. 4 <i>Mockup</i> Tampilan Halaman <i>Login</i>	18
Gambar 3. 5 <i>Mockup</i> tampilan halaman <i>Sign up</i>	19
Gambar 3. 6 <i>Mockup</i> tampilan <i>home dashboard</i> mahasiswa.....	20
Gambar 3. 7 <i>Mockup</i> tampilan sistem histori nilai kuis.....	20
Gambar 3. 8 <i>Mockup</i> tampilan sistem penyajian kuis	21
Gambar 3. 9 <i>Mockup</i> tampilan <i>home dashboard</i> dosen.....	22
Gambar 3. 10 <i>Mockup</i> tampilan <i>user management</i>	22
Gambar 3. 11 <i>Mockup</i> tampilan menambah kuis.....	23
Gambar 3. 12 Desain arsitektur back-end sitem	24
Gambar 4. 1 Tampilan <i>Login</i>	31
Gambar 4. 2 Tampilan <i>sign up</i>	32
Gambar 4. 3 Tampilan <i>home dashboard</i> mahasiswa	33
Gambar 4. 4 Tampilan histori nilai kuis mahasiswa.....	33
Gambar 4. 5 Tampilan <i>input</i> kode kuis.....	33
Gambar 4. 6 Tampilan penyajian kuis	34
Gambar 4. 7 Tampilan <i>home dashboard</i> dosen	34
Gambar 4. 8 Tampilan menambah kuis	35
Gambar 4. 9 Tampilan manajemen kuis	36
Gambar 4. 10 Tampilan menghapus kuis	36
Gambar 4. 11 Tampilan <i>user management</i>	37
Gambar 4. 12 Tampilan <i>leaderboard</i>	37



BAB I

PENDAHULUAN

1.1 Latar Belakang

Kemajuan teknologi informasi dan komunikasi telah membawa dampak yang besar pada berbagai bidang kehidupan, salah satunya pada bidang pendidikan atau pembelajaran di institusi pendidikan harus memikirkan cara pembelajaran yang sesuai dengan kebutuhan mahasiswa dan mengemas proses pembelajaran tersebut agar lebih bermakna, menarik, dan mengikuti perkembangan iptek, serta dapat membantu mahasiswa meningkatkan Indeks prestasi mereka. Suatu kegiatan yang sangat menyenangkan bagi mahasiswa yang biasa digunakan dosen dengan memberikan pertanyaan singkat pada saat kegiatan perkuliahan akan dimulai. Kegiatan ini disebut dengan kuis. Dalam perkembangannya, kuis dibuat untuk membantu kita dalam mengasah otak dan memikirkan berbagai strategi atau cara untuk menyelesaikan suatu masalah (Tanwir et al., 2018). Adanya perkembangan teknologi yang sangat pesat di era globalisasi dapat berpengaruh terhadap proses pembelajaran di kampus. Pembelajaran dalam kelas selama ini hanya terbatas pada penyampaian teori melalui metode ceramah dan hal ini berimbas pada hasil belajar yang diraih mahasiswa. Rendahnya hasil belajar mahasiswa terlihat dari indeks prestasi yang dicapai oleh mahasiswa (Ngafifi, 2014).

Walaupun nilai yang dihasilkan dari kuis tidak banyak, kuis telah menjadi opsi terbaik bagi dosen untuk memberi nilai tambahan bagi mahasiswa dengan memanfaatkan nilai dari setiap kuis yang biasa dilaksanakan tiap pertemuan kuliah, hal ini disadari membantu mahasiswa yang nilai perkuliahan di dalam kelas kurang memuaskan dan akhirnya menimbulkan dorongan bagi mahasiswa untuk belajar secara mandiri di luar kegiatan perkuliahan yang dilaksanakan di kelas. Diharapkan, ketika mahasiswa mendapatkan kuis dari dosen dalam mata kuliah tertentu, mahasiswa akan terbiasa untuk mengulang materi yang sebelumnya sudah pernah disampaikan di dalam perkuliahan.

Linked data merupakan suatu metode dalam menampilkan informasi dengan menggabungkan data dari berbagai sumber data di internet agar pengguna satu sama lain dapat memperkaya informasi data yang didapatkan serta mempermudah pencarian informasi data yang terkait dengan data yang sebelumnya telah diperoleh.

Pemanfaatan *Linked data* dapat menjadi solusi alternatif sebagai metode dalam memperoleh sumber data pada aplikasi penghasil kuis otomatis dengan sumber daya utama berbasis web semantik dalam mendapatkan ilmu ataupun materi yang ingin dijadikan topik kuis yang akan diberikan yang kerangka metadatanya dimodelkan dalam bentuk *Resource Description Framework* (RDF) (Miao et al., 2015) serta menjadikannya antarmuka yang intuitif dengan sistem yang akan dibuat untuk meningkatkan minat mahasiswa selaku target pengguna yang dituju.

12 Rumusan Masalah

Berdasarkan latar belakang diatas, maka dapat disimpulkan beberapa permasalahan yang mencakup hal-hal sebagai berikut,yaitu:

1. Bagaimana cara membangun suatu aplikasi yang dapat meghasilkan kuis secara otomatis sesuai topik yang diinginkan?
2. Bagaimana memanfaatkan konsep *linked data* sebagai sumber data utama dalam mendapatkan materi yang digunakan sebagai kuis?

13 Batasan Masalah

Batasan masalah pada penelitian ini dapat diuraikan sebagai berikut:

1. Membangun aplikasi dan *framework* apa yang dapat digunakan untuk menghasilkan kuis dari beberapa domain dengan upaya pencarian resource yang minimum.
2. Mekanisme yang akan diperkenalkan untuk menghasilkan kuis sesuai topik yang diinginkan.
3. Merancang antarmuka yang bersifat intuitif yang disediakan untuk meningkatkan interaksi antarmuka dengan pengguna.

14 Tujuan Penelitian

Tujuan pada penelitian ini adalah untuk mengembangkan aplikasi penghasil kuis otomatis dengan konsep *linked data* sebagai sumber data.

15 Manfaat Penelitian

Manfaat yang didapatkan pada penelitian ini adalah sebagai berikut :

1. Bagi dosen dan mahasiswa

Sebagai sumber informasi dalam penerapan sistem penghasil kuis otomatis dengan metode *linked data* dan memberikan kemudahan serta meningkatkan efisiensi dosen dalam membuat kuis sesuai topik yang diinginkan dan menyajikan pada pembelajaran.

2. Bagi peneliti

Menambah pengetahuan mengenai pemanfaatan *linked data* khusus pada sistem penghasil kuis.

16 Metodologi Penelitian

Berikut adalah metodologi yang digunakan dalam penelitian ini sebagai berikut.

1. Analisis

Pada tahap ini akan dijabarkan segala kebutuhan system yang akan dibuat guna mengidentifikasi berbagai masalah sehingga dapat dilakukan penanggulangan dengan baik atas masalah yang akan muncul pada tahap proses pengembangan aplikasi selanjutnya.

2. Perancangan

Pada tahap ini hasil dari tahap analisis akan diimplementasikan dalam bentuk pemodelan kasus berupa pembuatan *use case diagram* dan *activity diagram* serta tampilan *mockup* aplikasi yang akan dibuat.

3. Pembuatan

Pada tahap ini sistem atau program mulai dibuat sesuai dengan racangan yang dibuat pada tahap sebelumnya.

4. Pengujian dan Evaluasi

Pengujian dilakukan guna memastikan program yang dijalankan sesuai dengan apa yang telah dirancang dan melihat adanya *bug* yang terjadi agar mengetahui kekurangan dan kelebihan dari sistem atau program yang telah dibuat dengan cara pengujian langsung terhadap fungsionalitas sistem yang telah dibuat. Hasil pengujian juga digunakan sebagai *insight* dari program yang telah dibuat untuk dikembangkan lebih lanjut .

17 Sistematika Penulisan

Sistematika Penulisan dibuat untuk membantu memberikan gambaran secara umum tentang penelitian yang akan dilakukan. Secara garis besar sistematika penulisan sebagai berikut:

BAB 1 PENDAHULUAN

Bab ini akan menjelaskan secara singkat mengenai latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian dan sistematika penulisan laporan TA.

BAB II LANDASAN TEORI

Bab kedua ini memuat perihal teori-teori yang digunakan dalam sistem, seperti pengertian, penjelasan, dan penelitian terdahulu.

BAB III METODOLOGI PENELITIAN

Bab ini memuat obyek penelitian, data yang digunakan serta tahapan pengembangan perangkat lunak yang meliputi rancangan antarmuka dan fungsionalitas aplikasi yang telah dilakukan dalam penelitian ini secara ringkas dan jelas.

BAB IV PEMBAHASAN

Bab ini memuat pembahasan dan hasil yang diperoleh dalam penelitian di mana kesesuaian hasil dengan tujuan penelitian seperti interface dan uraian tentang fitur aplikasi yang telah dibuat.

BAB V KESIMPULAN DAN SARAN

Bab ini berisi tentang kesimpulan yang diperoleh melalui pembahasan di bab sebelumnya. Kemudian saran dibuat berdasarkan pengalaman dan pertimbangan penulis yang ditujukan kepada para peneliti dalam bidang sejenis sebagai rekomendasi agar hasil penelitian tersebut dapat dilanjutkan.

BAB II

LANDASAN TEORI

21 SISTEM INFORMASI

Menurut (Kusnendi, 2014), sistem informasi adalah seperangkat komponen yang terintegrasi untuk mengumpulkan, menyimpan, dan memproses data dan guna menyediakan berbagai macam informasi, pengetahuan, dan produk digital. Perusahaan yang bergerak di bidang bisnis dan organisasi sekarang sangat bergantung pada sistem informasi. Output yang dihasilkan dari sistem informasi tersebut berguna dalam mendukung proses pengambilan keputusan dengan melakukan analisis atas permasalahan yang diapatkan (Anggadini, 2013).

22 WEB SEMANTIK

Web Semantik adalah bentuk web yang memiliki basis kecerdasan tertentu sehingga membuatnya berbeda dari web biasa pada umumnya. Web semantik memiliki kemampuan komputer dalam memahami kebutuhan pengguna menjadi lebih mudah. Web semantik dapat dijalankan dengan adanya sekumpulan standar yang dikoordinasi oleh *World Wide Web Consortium (W3C)*. Kombinasi paling penting sebagai standar dalam membangun web semantik adalah XML, XML *Schema*, RDF, dan SPARQL. Salah satu contoh penggunaan Web semantik adalah pada saat web *e-commerce* dapat menampilkan rekomendasi produk kepada pengguna berdasarkan aktivitas yang biasa dilakukan atau dicari pada laman web tersebut (Dwiono, 2013).

23 LINKED DATA

Linked Data adalah istilah dalam ilmu komputer yang merujuk pada metode publikasi data yang terstruktur. Data-data ini saling terkait, sehingga lebih bermanfaat bagi mereka yang ingin berkonsultasi dan memperdalam topik tertentu. *Linked Data* muncul dari proyek Semantic Web, yang ingin mempromosikan pengenalan metadata semantik di *World Wide Web* (Heath & Bizer, 2011). *Linked data* memiliki peran penting dalam pembangunan web

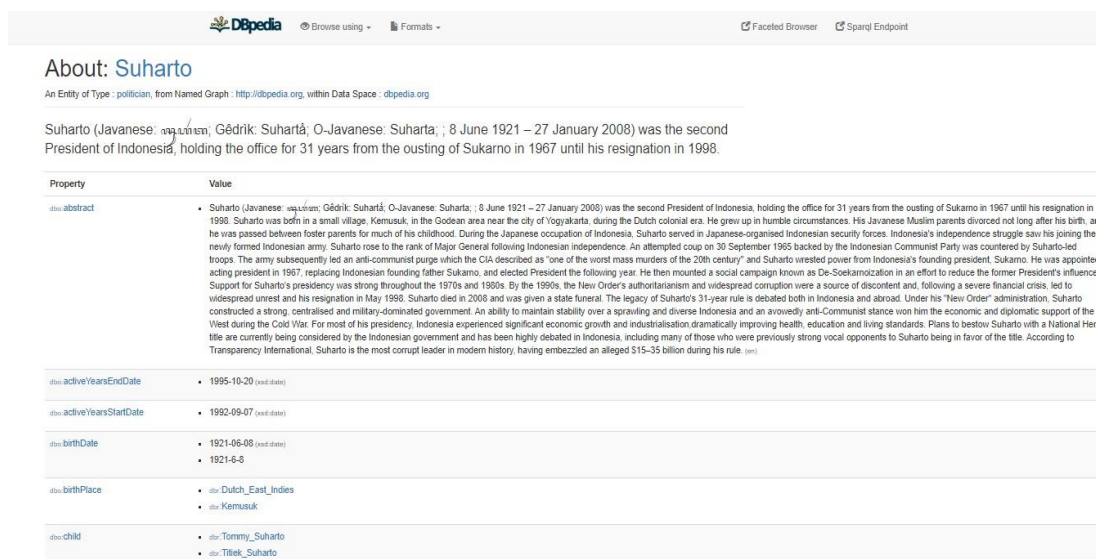
semantik. Menurut (Bizer et al., 2009), pemanfaatan konsep *linked data* itu sendiri memiliki empat prinsip dasar yaitu.

1. Penggunaan URI guna mengidentifikasi dan pemberian nama unik pada berbagai source dari konten digital yang tersedia di pada internet dari objek dunia nyata hingga konsep abstrak. Dengan bantuan URI, kita dapat membedakan berbagai hal atau mengetahui bahwa satu dataset sama atau berbeda dengan dataset yang lain.
2. Penggunaan protokol HTTP pada URI agar pengguna dapat mudah dalam mengaksesnya di internet.
3. Penggunaan SPARQL dan RDF sebagai standar dalam pengambilan dan penyajian informasi secara detil pada URI yang dicari.
4. Menyediakan *link* atau tautan pada URI yang sudah ada agar pengguna mudah mendapatkan informasi yang saling terkait URI sebelumnya.

Pemanfaatan Linked data diawali dengan proyek Linking Open Data yang diprakarsai oleh *W3C Semantic Web Education and Outreach (SWEO) Interest Group*. Proyek ini bertujuan untuk memperkenalkan konsep dasar dari pemanfaatan *Linked data* itu sendiri dengan cara membagikan berbagai macam *open data set* dalam format RDF dan mengatur tautan-tautan RDF di internet dari berbagai sumber data. Sampai September 2011, proyek tersebut telah mempublikasikan dan menghubungkan lebih dari 31 miliar RDF *triple data set* yang saling terkait yang juga disebut sebagai “*LOD cloud*”.

24 DBPEDIA

DBpedia adalah basis pengetahuan berskala besar dan multibahasa yang dihasilkan dengan mengekstraksi data terstruktur dari *Wikipedia*. Ada beberapa upaya untuk menggunakan *DBpedia* untuk menghasilkan pertanyaan untuk sebuah kuis, tetapi inisiatif ini belum berhasil menghasilkan set pertanyaan yang besar, beragam, dan menghibur. Selain itu, masalah latensi terlalu tinggi untuk kuis yang interaktif jika *query* pertanyaan yang dibuat dengan mengirimkan pertanyaan langsung ke titik akhir yaitu *DBpedia* publik (Auer et al., 2007). Gambar 2.1 menunjukkan tampilan *Dbpedia* dalam menampilkan query pencarian Presiden Suharto.



About: Suharto
An Entity of Type `politician`, from Named Graph `http://dbpedia.org`, within Data Space `dbpedia.org`

Suharto (Javanese: ꦱꦸꦲꦂꦠꦺ, Gèdrik: Suhartá, O-Javanese: Suharta; ; 8 June 1921 – 27 January 2008) was the second President of Indonesia, holding the office for 31 years from the ousting of Sukarno in 1967 until his resignation in 1998.

Property	Value
<code>dbpedia:abstract</code>	<ul style="list-style-type: none"> Suharto (Javanese: ꦱꦸꦲꦂꦠꦺ, Gèdrik: Suhartá, O-Javanese: Suharta; ; 8 June 1921 – 27 January 2008) was the second President of Indonesia, holding the office for 31 years from the ousting of Sukarno in 1967 until his resignation in 1998. Suharto was born in a small village, Kemusuk, in the Godean area near the city of Yogyakarta, during the Dutch colonial era. He grew up in humble circumstances. His Javanese Muslim parents divorced not long after his birth, and he was passed between foster parents for much of his childhood. During the Japanese occupation of Indonesia, Suharto served in Japanese-organised Indonesian security forces. Indonesia's independence struggle saw his joining the newly formed Indonesian army. Suharto rose to the rank of Major General following Indonesian independence. An attempted coup on 30 September 1965 backed by the Indonesian Communist Party was countered by Suharto-led troops. The army subsequently led an anti-communist purge which the CIA described as "one of the worst mass murders of the 20th century" and Suharto wrested power from Indonesia's founding president, Sukarno. He was appointed acting president in 1967, replacing Indonesian founding father Sukarno, and elected President the following year. He then mounted a social campaign known as De-Soekarnozation in an effort to reduce the former President's influence. Support for Suharto's presidency was strong throughout the 1970s and 1980s. By the 1990s, the New Order's authoritarianism and widespread corruption were a source of discontent and, following a severe financial crisis, led to widespread unrest and his resignation in May 1998. Suharto died in 2008 and was given a state funeral. The legacy of Suharto's 31-year rule is debated both in Indonesia and abroad. Under his 'New Order' administration, Suharto constructed a strong, centralised and military-dominated government. An ability to maintain stability over a sprawling and diverse Indonesia and an avowedly anti-Communist stance won him the economic and diplomatic support of the West during the Cold War. For most of his presidency, Indonesia experienced significant economic growth and industrialisation dramatically improving health, education and living standards. Plans to bestow Suharto with a National Hero title are currently being considered by the Indonesian government and has been highly debated in Indonesia, including many of those who were previously strong vocal opponents to Suharto being in favor of the title. According to Transparency International, Suharto is the most corrupt leader in modern history, having embezzled an alleged \$15–35 billion during his rule. (en)
<code>dbpedia:activeYearsEndDate</code>	<ul style="list-style-type: none"> 1995-10-20 (xsd:date)
<code>dbpedia:activeYearsStartDate</code>	<ul style="list-style-type: none"> 1982-09-07 (xsd:date)
<code>dbpedia:birthDate</code>	<ul style="list-style-type: none"> 1921-06-08 (xsd:date) 1921-6-8
<code>dbpedia:birthPlace</code>	<ul style="list-style-type: none"> <code>dbpedia:Dutch_East_Indies</code> <code>dbpedia:Kemusuk</code>
<code>dbpedia:child</code>	<ul style="list-style-type: none"> <code>dbpedia:Tommy_Suharto</code> <code>dbpedia:Tilik_Suharto</code>

Gambar 2. 1 tampilan DBpedia.org

25 RDF

RDF adalah komponen penting dalam membangun web semantic. Suatu kerangka kerja untuk merepresentasikan informasi di Web. Konsep dan Abstrak *Syntax* RDF mendefinisikan sintaksis abstrak yang menjadi dasar RDF, dan yang berfungsi untuk menghubungkan sintaks konkretnya dengan semantic formal. Ini juga mencakup diskusi tentang tujuan desain, konsep-konsep utama, tipe data, normalisasi karakter dan penanganan referensi URI (Klyne & Carroll, 2004). Menurut (Ibrahim, 2007) konsep dasar dari pembuatan RDF itu sendiri adalah tentang cara mendeklarasikan sebuah resource pada internet dalam bentuk “Subjek-Predikat-Objek” atau dalam RDF sendiri disebut dengan istilah format *N-triple*. Sebagai contoh dalam merepresentasikan suatu kalimat “W.R Soepratman adalah pencipta lagu Indonesia Raya” dalam format *N-triple*, maka dalam S-P-O, yang bertindak sebagai subjek adalah “W.R Soepratman, predikatnya “ adalah pencipta lagu”, dan objeknya adalah lagu Indonesia Raya.

Agar RDF dapat dipahami oleh komputer maka diperlukan sebuah *identifier*. Nantinya *identifier* ini akan diberikan pada entitas yang memiliki informasi yang saling berkaitan. Bentuk identifier pada RDF adalah URI (uniform resource identifier). Jika mengambil contoh kalimat “W.R Soepratman adalah pencipta lagu Indonesia Raya” maka penulisan tautannya adalah `<http://example.org/#Soepratman><http://www.perceive.net/schemas/relationship/penciptaDari>` `<http://example.org/#LaguIndonesiaraya>` yang dapat diuraikan sebagai.

1. <http://example.org/#Soepratman> sebagai subjek yang melambangkan *resource* berupa URL.

2. <http://www.perceive.net/schemas/relationship/penciptaDari> sebagai predikat.
3. <http://example.org/#LaguIndonesiaraya> sebagai objek.

Selain menggunakan format N-triple, pernyataan tersebut dapat dideklarasikan dengan menggunakan format RDF sebagai berikut:

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://www.perceive.net/schemas/relationship/penciptaDari/">
  <rdf:Description
rdf:about="http://example.org/#Soepratman">
  <dc:title>Jadwal Ujian</dc:title>
  </rdf:Description>
</rdf:RDF>
```

26 SPARQL

Simple Protocol dan *RDF Query Language* atau biasa disingkat SPARQL adalah bahasa untuk memanggil RDF yang merupakan rekomendasi W3C. SPARQL berisi kemampuan untuk query pola grafik yang diperlukan dan sering digunakan karena kemampuan konjungsi dan disjungsi mereka (Barbieri et al., 2009), kemudian dari pola grafik tersebut akan dicocokkan dengan berbagai *endpoint* dari *repository* yang dituju untuk mencari informasi yang dicari. Standar SPARQL untuk mendefinisikan query pencarian dan bahasa mirip seperti bahasa SQL. Hasil dari query pencarian yang digunakan untuk melakukan pengambilan data berbentuk RDF maupun XML. SPARQL dianggap setara dengan bahasa SQL, namun struktur SPARQL lebih sederhana dan gampang untuk diimplementasikan khususnya dalam pengembangan web semantik.

Berikut adalah contoh dalam menggunakan query dalam mengambil contoh informasi dari Dbpedia.org mengenai ibukota provinsi Kalimantan Timur dan etnis suku yang ada di Kalimantan Timur.

```
select ?ibukota ?etnis
  where {
    dbpedia-id:Kalimantan_Timur dbpedia-owl:capital ?ibukota.
    dbpedia-id:Kalimantan_Timur dbpedia-owl:ethnicGroup ?etnis.
  }
```

Hasil query berupa bentuk tabel dalam bentuk HTML atau dalam format RDF/XML. Hasil query dalam bentuk tabel disesuaikan dengan subjek, predikat dan objek yang ditunjukkan pada

Gambar 2.2 . Menurut (W3C, 2008) , SPARQL memiliki 6 variasi query yang berbeda-beda yang memiliki fungsi yang berbeda juga, diantaranya adalah :

1. PREFIX

PREFIX digunakan untuk meringkas sebuah resource, dalam hal ini dapat diwakili oleh URI (*Uniform Resource Identifier*).

2. SELECT

Mengembalikan keseluruhan atau sekumpulan dari variabel yang diikat di dalam pola dari suatu *query*.

3. CONSTRUCT

Mengembalikan grafik RDF yang dikonstruksioleh variabel pengganti dalam sekumpulan format *N-triple*.

4. ASK

Mengembalikan tipe data yang menunjukkan pola dari query cocok atau tidak.

5. OPTIONAL

Digunakan untuk mengatasi ketidakcocokan struktur pola query dengan pola yang ada pada graf RDF.

6. DESCRIBE

menampilkan sumber data yang dipanggil.

ibukota	etnis
http://id.dbpedia.org/resource/Samarinda	http://id.dbpedia.org/resource/Suku_Banjar
http://id.dbpedia.org/resource/Samarinda	http://id.dbpedia.org/resource/Suku_Bugis
http://id.dbpedia.org/resource/Samarinda	http://id.dbpedia.org/resource/Tionghoa-Indonesia
http://id.dbpedia.org/resource/Samarinda	http://id.dbpedia.org/resource/Suku_Jawa
http://id.dbpedia.org/resource/Samarinda	http://id.dbpedia.org/resource/Suku_Madura
http://id.dbpedia.org/resource/Samarinda	http://id.dbpedia.org/resource/Suku_Sunda
http://id.dbpedia.org/resource/Samarinda	http://id.dbpedia.org/resource/Suku_Toraja
http://id.dbpedia.org/resource/Samarinda	http://id.dbpedia.org/resource/Suku_Kutai
http://id.dbpedia.org/resource/Samarinda	http://id.dbpedia.org/resource/Suku_Kutai

Gambar 2. 2 hasil tabel format HTML pencarian *query* dengan SPARQL

27 FLASK

Flask adalah sebuah *platform web* khusus dibuat dan dapat digunakan dalam bahasa pemrograman *Python*. Berbeda dengan platform web framework lainnya, *Flask* sangat cocok digunakan oleh *web developer* pemula yang ingin membangun dan mempelajari pembuatan web karena sifatnya yang *microframework* yaitu berfokus kepada kebutuhan pokok yang

dibutuhkan dalam pembuatan aplikasi *web* dan juga digunakan dalam proyek skala kecil , oleh karena itu *framework Flask* sangat cocok digunakan oleh web *developer* pemula karena tampilan serta *back-end* suatu web dapat lebih mudah diatur karena *Flask* menggunakan bantuan aplikasi pihak ketiga dan dapat menggunakannya sebagai ekstensi dan *library package* yang membuatnya sebagai web framework dengan fleksibilitas dan skalabilitas yang tinggi (Grinberg, 2014).

28 NLTK

Natural Language Toolkit atau biasa disingkat NLTK adalah pustaka sumber terbuka untuk bahasa pemrograman *Python* yang awalnya ditulis oleh Steven Bird, Edward Loper dan Ewan Klein untuk digunakan dalam pengembangan dan pendidikan. Dengan NLTK kita dapat mempelajari dasar-dasar pemrograman linguistik serta *text processing* agar mesin atau komputer dapat memahami input dengan bahasa manusia dan melakukan respon yang sesuai. Berbagai macam dasar proses text processing yang dapat dilakukan dengan NLTK adalah tokenisasi kata, filterisasi kalimat menjadi kata, pemisahan kata dari imbuhan dan penghitungan jumlah kata atau karakter dalam suatu teks (Bird et al., 2009).

29 PENELITIAN TERKAIT

Bab ini akan memaparkan beberapa literatur penelitian terdahulu yang memiliki keterkaitan dengan penelitian dan sistem yang akan dibuat. Penelitian dengan judul Penghasil Kuis dengan Konsep Linked Data yang akan dikembangkan dalam bentuk *website*.

Beberapa literatur penelitian terdahulu telah didapatkan diantaranya Penelitian yang dilakukan oleh Pambudi & Sismoro pada tahun 2013 mengenai pembuatan aplikasi kuis namun dengan *platform* android. Penelitian sebelumnya juga dilakukan oleh Vega-Gorgojo dan Liu & Lin pada tahun 2019 dan 2014 mengenai konsep awal bagaimana pemanfaatan konsep linked data pada aplikasi penghasil kuis. Keterkaitan beberapa literatur tersebut dapat dilihat pada Tabel 2.1 di bawah.

Tabel 2. 1 Penelitian Terkait

Judul	Keterkaitan dengan Sistem yang akan dibuat	Sumber Literatur
Pembuatan Aplikasi Kuis Perangkat Komputer Berbasis Android	Memiliki kesamaan dalam menjelaskan penelitian tentang aplikasi penghasil quiz namun,peneliti tidak memanfaatkan konsep linked data dan mengguakan Bahasa pemrograman android	(Pambudi & Sismoro, 2013)
Clover Quiz: a trivia game	Menjelaskan berbagai macam penelitian aplikasi	(Vega-

Judul	Keterkaitan dengan Sistem yang akan dibuat	Sumber Literatur
powered by DBpedia	penghasil quiz otomatis dengan memanfaatkan konsep linked data	Gorgojo, 2019)
Sherlock: a Semi-Automatic Quiz Generation System using Linked Data	Menjelaskan alasan mengapa konsep linked data dapat cocok digunakan sebagai suatu metode dalam menggali, mendapatkan, mengekstrak, dan mempublish data ke dalam web, secara otomatis, dan khususnya dalam konsep pembuatan aplikasi penghasil quiz otomatis	(Liu & Lin, 2014)



BAB III METODOLOGI PENELITIAN

31 Lokasi dan Waktu penelitian

Penelitian ini dilakukan di Kampus Terpadu Universitas Islam Indonesia Yogyakarta di Fakultas Teknik Industri, Jurusan Informatika. Penelitian ini dimulai pada Tanggal 5 Mei 2019 hingga 1 Juni 2020.

32 Alat Penelitian

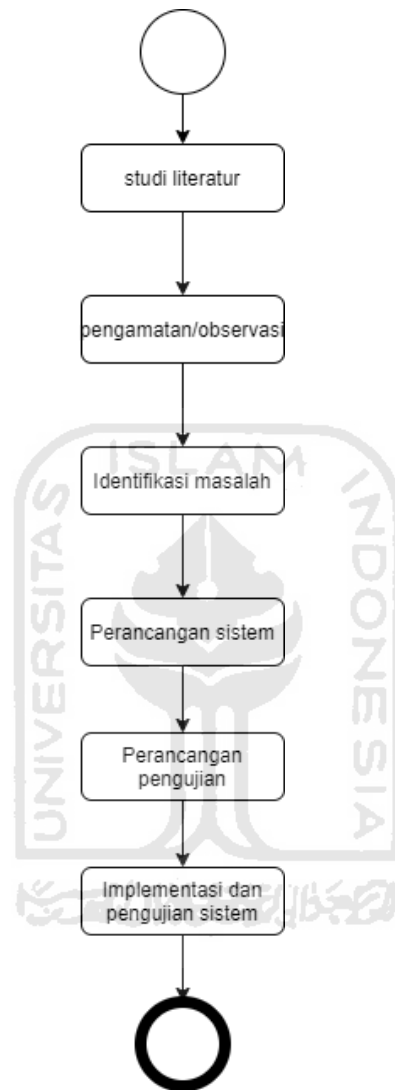
Pada penelitian ini akan membutuhkan peralatan berupa hardware meliputi perangkat keras yang membantu dalam proses pembuatan aplikasi serta sebagai alat uji guna mengetahui indikator kekurangan yang harus dilengkapi pada sistem yang akan dibuat.

Tabel 3. 1 Daftar peralatan penelitian

No	Alat	Spesifikasi
1	Laptop	Asus X550J
2	CPU	Intel <i>Skylake Core</i> i7-4720HQ
3	VGA	GTX 950M 2GB DDR3
4	<i>Memory</i>	4GB up to 12GB DDR3L 1600
5	<i>Storage</i>	500GB SATA 5400RPM

33 Metode Penelitian

Metode penelitian adalah serangkaian cara yang digunakan oleh peneliti dalam melakukan penyusunan laporan. Tahapan penelitian tersebut dapat diuraikan pada gambar 3.1 yang meliputi tahap studi literatur berikut.



Gambar 3. 1 *Flowchart* tahapan metode penelitian

34 Studi Literatur

1. Metode Studi Literatur

Studi literatur dilakukan dengan mempelajari sejumlah referensi yang berkaitan erat dengan materi penelitian baik melalui *website google scholar* dan *science direct*.

2. Perancangan dan analisa Sistem

Dengan mengidentifikasi karakteristik sistem yang sudah ada berdasarkan literatur yang di dapatkan serta kriteria yang dibutuhkan dalam penelitian ini guna mendapatkan gambaran umum tentang aplikasi yang akan dibuat.

3. Penulisan TA

Meliputi pembahasan dari analisis sistem hingga pembuatan aplikasi berbasis teknologi linked data untuk yang menjadi dasar dalam penulisan skripsi (TA).

35 Identifikasi masalah

Identifikasi masalah dilakukan dengan observasi dan melakukan wawancara kepada pihak dosen sebagai. Pada saat melakukan wawancara, penulis mengajukan pertanyaan kepada pihak terkait guna mendapatkan informasi, kebutuhan sistem dari dosen untuk digunakan pada sistem. Informasi yang didapatkan akan diidentifikasi sebagai bahan pendukung dalam implementasi sistem yang akan dikerjakan.

36 Analisis Kebutuhan

Analisis kebutuhan bertujuan untuk memahami kebutuhan suatu sistem. Analisis kebutuhan meliputi dua hal yaitu dari sisi fungsional dan non-fungsional

3.6.1 Kebutuhan fungsional

Analisis dalam kebutuhan fungsional pada system yang akan dibuat bermanfaat dalam menjelaskan kebutuhan apa saja yang meliputi serangkaian proses yang akan dilakukan oleh sistem yang akan dibuat. Berikut adalah Tabel 3.2 yang menunjukkan kebutuhan fungsional yang harus ada dalam sistem yang akan dibuat yang didapatkan berdasarkan hasil observasi dan wawancara kepada pihak dosen.

Tabel 3. 2 Daftar prioritas kebutuhan fungsional

No	Deskripsi kebutuhan	Prioritas
1	Sistem mampu menampilkan menu utama.	Harus ada.
2	Sistem dapat menghasilkan dan menampilkan soal sesuai input topik yang dipilih.	Harus ada
3	Sistem mampu menampilkan soal berdasarkan jumlah soal yang diinputkan.	Harus ada.

4	Sistem mampu menampilkan jawaban benar.	Harus ada.
5	Sistem mampu menampilkan jawaban salah.	Harus ada.
6	Sistem mampu menampilkan nilai tertinggi dan terendah.	Harus ada.

3.6.2 Kebutuhan non-fungsional

Analisis dalam kebutuhan non fungsional bermanfaat dalam menentukan kebutuhan spesifikasi sistem yang meliputi perangkat lunak yang membantu dalam proses pembuatan aplikasi serta sebagai indikator kekurangan yang harus dilengkapi pada sistem yang akan dibuat. Tabel 3.3 menunjukkan daftar kebutuhan non-fungsional dari sistem yang akan dibuat.

Tabel 3. 3 Kebutuhan non-fungsional perangkat lunak sistem

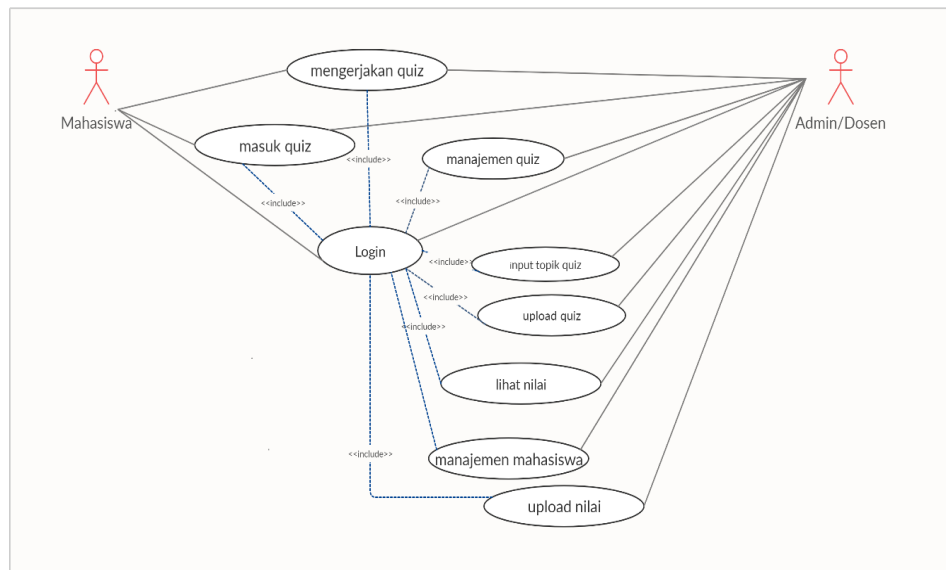
No	Aspek	Spesifikasi
1	<i>Platform</i>	<i>Web application</i>
2	Sistem operasi	<i>Windows</i>
3	Bahasa pemrograman	<i>Python</i>
4	<i>Framework</i>	<i>Flask</i>
5	Ukuran File	Ringan < 100mb

37 Rancangan Aplikasi

Pada tahap rancangan aplikasi ini, gambaran umum sistem dari tahap analisis yang dimodelkan dalam *UML* berupa pembuatan *use case diagram* dan *activity diagram* serta *mockup* tampilan yang akan dibuat.

3.7.1 Use Case Diagram

Use Case Diagram bermanfaat dalam memetakan kebutuhan fungsional sistem. Berikut Gambar 3.1 di bawah menunjukkan pemodelan *Use Case Diagram* sistem yang akan dibuat.



Gambar 3. 2 *Use Case Diagram* Pengembangan Aplikasi Penghasil Kuis Otomatis

Gambar 3.2 menunjukkan *Use Case Diagram* yang merepresentasikan jumlah aktor dalam sistem yang akan dibuat yaitu mahasiswa dan dosen yang dimana setiap aktor memiliki hak yang berbeda pada sistem yang akan dibuat. Berikut penjelasannya antara lain :

A. Dosen

Dosen memiliki seluruh hak akses pada sistem yang akan dibuat meliputi :

1. *Login* ke sistem.
2. Manajemen kuis (mengatur jumlah soal dan batas akhir waktu pengumpulan kuis).
3. Input topik kuis.
4. Upload kuis.
5. Lihat nilai.
6. Upload nilai.
7. Manajemen mahasiswa(mengurangi atau menambah jumlah mahasiswa yang boleh mengerjakan kuis).

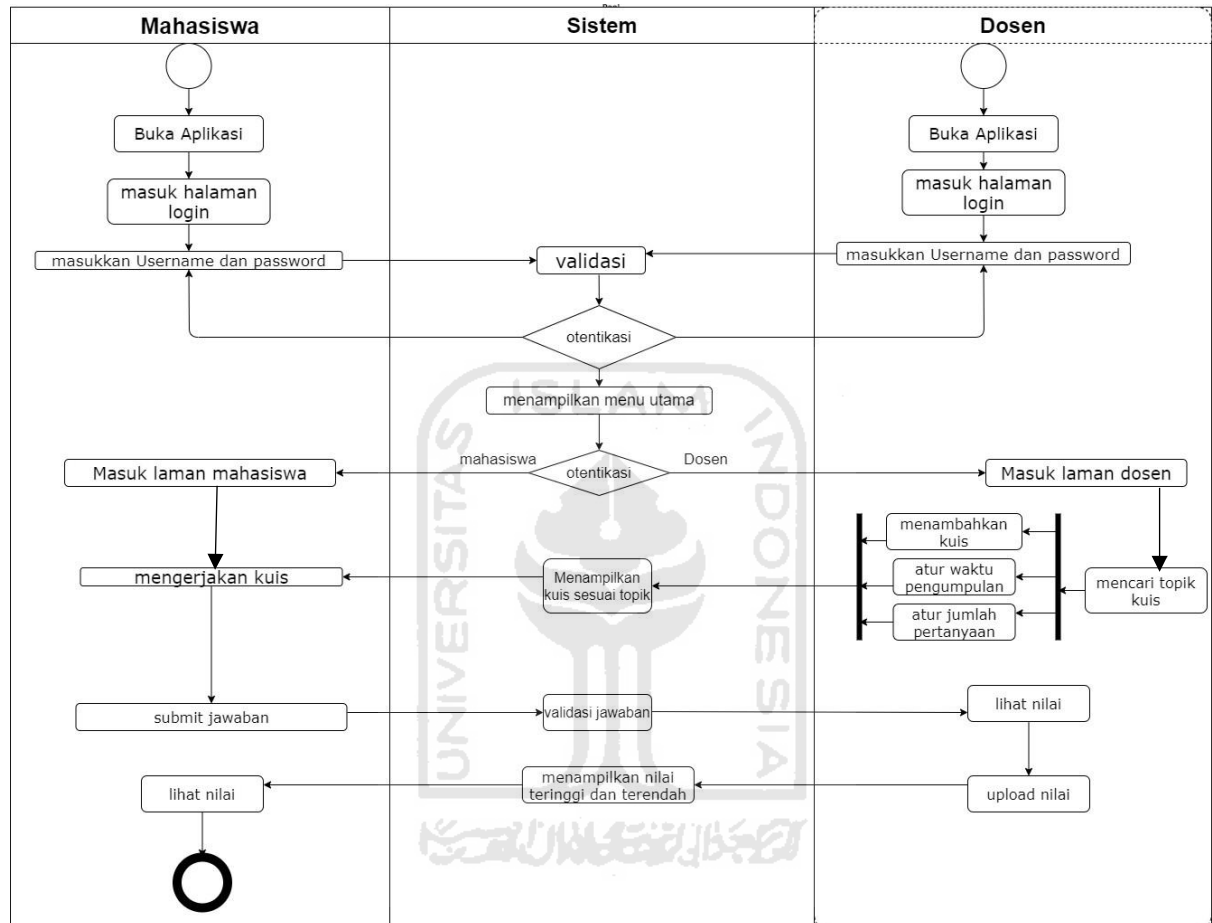
B. Mahasiswa

Mahasiswa memiliki 3 hak akses pada sistem yang akan dibuat meliputi :

1. *Login* ke sistem.
2. Masuk ke kuis yang telah di upload.
3. Mengerjakan kuis.
4. Melihat nilai.

3.7.2 Activity Diagram

Untuk mengetahui urutan aktifitas yang terjadi pada sistem yang telah digambarkan dalam *use case diagram* dapat dilihat dalam rancangan *activity diagram* pada gambar 3.3 di bawah ini.



Gambar 3. 3 Activity Diagram Pengembangan Aplikasi Penghasil Kuis Otomatis

Pada Gambar 3.3 menjelaskan bahwa dosen dan mahasiswa memiliki hak akses untuk masuk pada sistem, namun untuk dosen masuk pada laman yang berbeda. Dosen memiliki hak akses meliputi melakukan input kuis yang akan dikerjakan sesuai topik yang ditentukan, melakukan upload kuis untuk dikerjakan, serta memiliki hak akses dalam manajemen mahasiswa seperti menambah dan mengurangi jumlah mahasiswa yang dapat mengakses kuis yang telah dibuat pada sistem. Mahasiswa dapat masuk pada laman mahasiswa untuk mengerjakan kuis yang telah dibagikan secara otomatis ketika dosen membuat kuis dengan topik baru. Mahasiswa dapat memulai mengerjakan soal dan memilih jawaban dari soal yang disediakan berupa

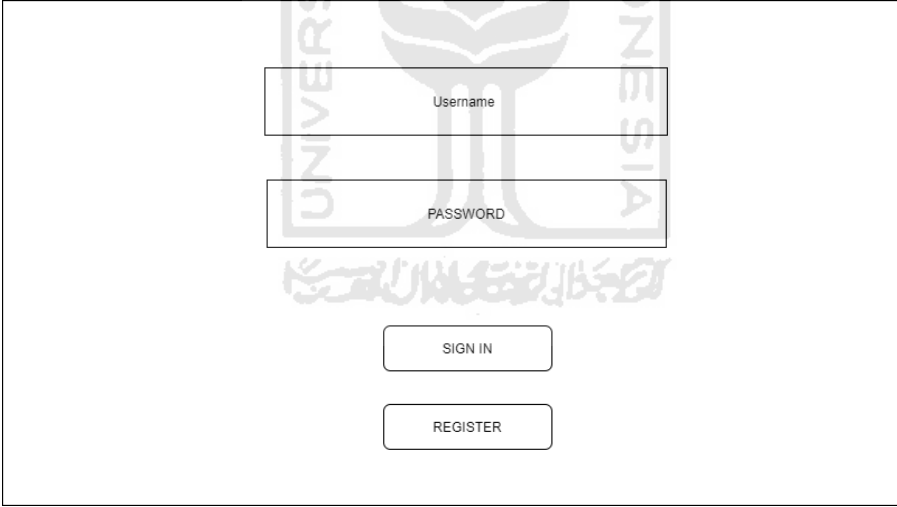
pilihan ganda. Ketika mahasiswa telah memilih jawaban, maka mahasiswa diharuskan untuk melakukan pengumpulan.

3.7.3 *Mockup sistem*

Pembuatan *mockup* bertujuan untuk menggambarkan desain awal antarmuka sistem yang akan diimplementasikan. Beberapa *mockup* sistem yang sudah dibuat dapat dilihat sebagai berikut.

3.7.3.1 *Rancangan halaman login*

Gambar 3.4 merupakan rancangan antarmuka halaman *login* bagi dosen atau mahasiswa untuk dapat mengakses sistem menuju halamannya masing-masing dengan cara memasukkan *username* dan *password*.



The image shows a login form mockup. It features two input fields: the top one is labeled 'Username' and the bottom one is labeled 'PASSWORD'. Below these fields are two buttons: 'SIGN IN' and 'REGISTER'. The entire form is centered on a white background. A large, faint watermark of the Universitas Islam Indonesia logo is visible in the background of the form area.

Gambar 3. 4 *Mockup* Tampilan Halaman *Login*

3.7.3.2 *Rancangan halaman sign up*

Pada Gambar 3.5 merupakan rancangan antarmuka halaman *sign up* bagi mahasiswa yang belum terdaftar pada sistem yang akan dibuat dengan mengisi form yang tersedia yaitu, *username* dan *password* kemudian meng-klik tombol *sign up* dan kembali ke halaman login. Mahasiswa yang sudah mendaftar pada *form sign up* bisa masuk ke dalam

sistem dengan mengisi *form login* yang kosong sesuai isian yang dimasukkan pada *form sign up*.



The image shows a registration form layout. It consists of four input fields and one button. The first row has two side-by-side fields: 'NAMA DEPAN*' and 'NAMA BELAKANG*'. The second row has a single wide field for 'USERNAME*'. The third row has a single wide field for 'PASSWORD*'. The fourth row has a single wide field for 'CONFIRM PASSWORD*'. Below these fields is a 'REGISTER' button. A faint watermark of the Universitas Islam Indonesia logo is visible in the background.

Gambar 3. 5 *Mockup* tampilan halaman *Sign up*

3.7.3.3 Rancangan halaman *dashboard* mahasiswa

Pada Gambar 3.6 menunjukkan rancangan tampilan home dari halaman mahasiswa. Disini kita dapat melihat kuis beserta topik yang sudah di *upload* oleh dosen melalui sistem yang ada serta terdapat beranda yang berisi informasi kuis yang akan datang serta kapan tanggal dan waktu mahasiswa mulai mengerjakan kuis dan batas akhir pengumpulan kuis. Pada Gambar 3.7 kita juga dapat melihat tampilan beranda untuk melihat histori kuis yang sudah pernah dikerjakan sebelumnya dan rincian nilai yang didapatkan. Kemudian untuk Gambar 3.8 adalah rancangan tampilan bagaimana kuis disajikan yaitu berbentuk pilihan ganda dan untuk menjawab kuis mahasiswa cukup dengan meng-klik *radio button* yang dirasa benar.



Gambar 3. 6 *Mockup* tampilan *home dashboard* mahasiswa



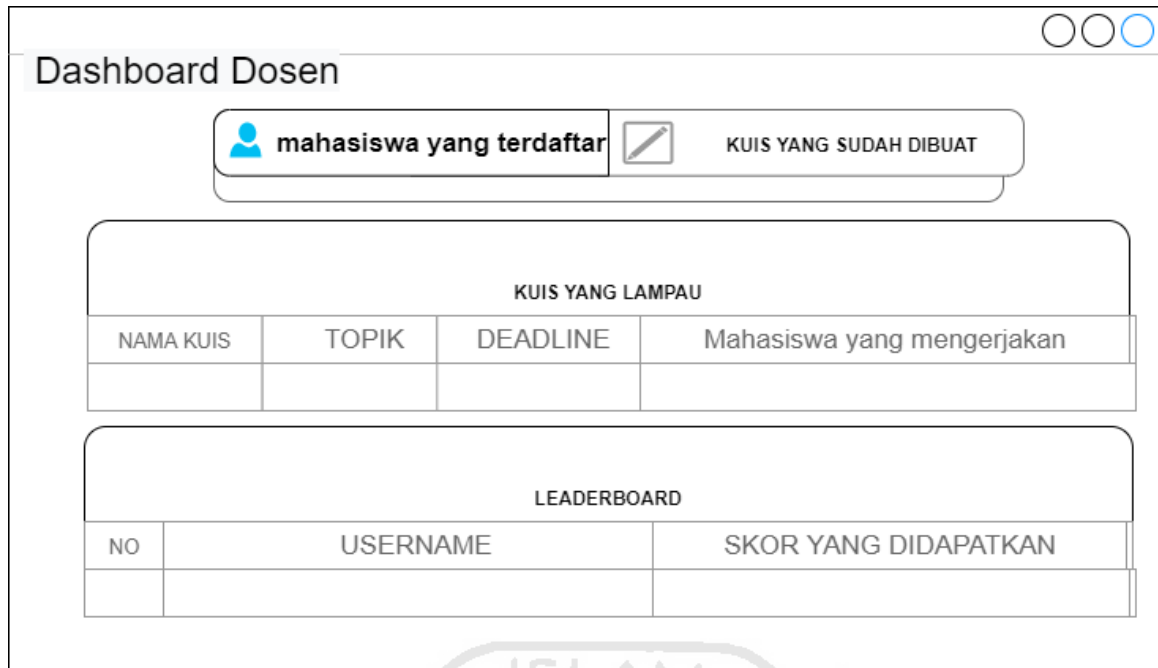
Gambar 3. 7 *Mockup* tampilan sistem histori nilai kuis



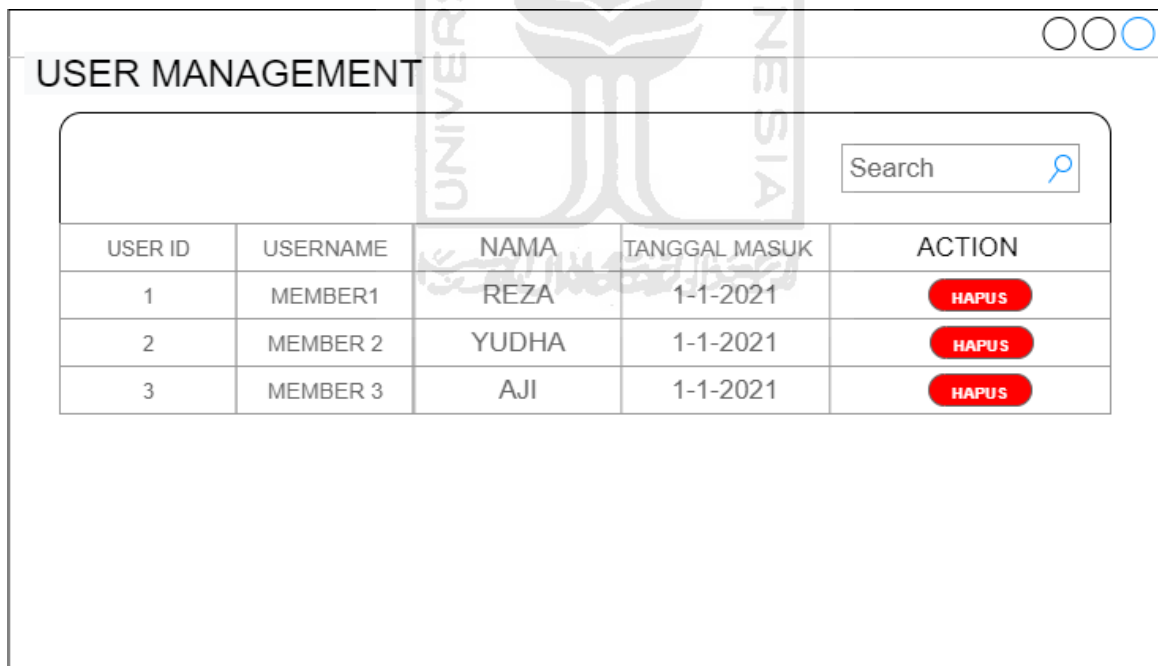
Gambar 3. 8 *Mockup* tampilan sistem penyajian kuis

3.7.3.4 Rancangan halaman dashboard dosen

Pada gambar 3.9 menampilkan rancangan tampilan *home dashboard* dosen yang mempunyai segala hak akses pada sistem yang akan dibuat. Disini dosen dapat melihat kuis beserta topik yang sudah di *input* melalui sistem yang ada. Pada gambar 3.9 juga menampilkan tampilan beranda untuk melihat histori kuis yang sudah pernah dibuat sebelumnya dan tabel *leaderboard* untuk mengetahui nilai terendah dan tertinggi dari mahasiswa yang telah mengerjakan kuis. Kemudian untuk gambar 3.10 adalah rancangan tampilan *user management* untuk menambah dan mengurangi hak akses mahasiswa yang boleh masuk ke sistem dan mengerjakan kuis. Gambar 3.11 adalah rancangan tampilan untuk melakukan *query* pencarian terhadap topik kuis yang diinginkan, mengatur waktu *deadline* pengumpulan kuis, dan jumlah soal yang akan ditampilkan.



Gambar 3. 9 *Mockup* tampilan *home dashboard* dosen



Gambar 3. 10 *Mockup* tampilan *user management*

Menambah Kuis

Nama Kuis

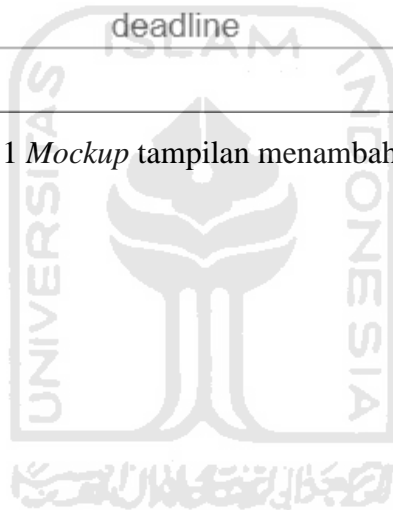
Search
Topik yang diinginkan

jumlah soal 100

🕒 deadline

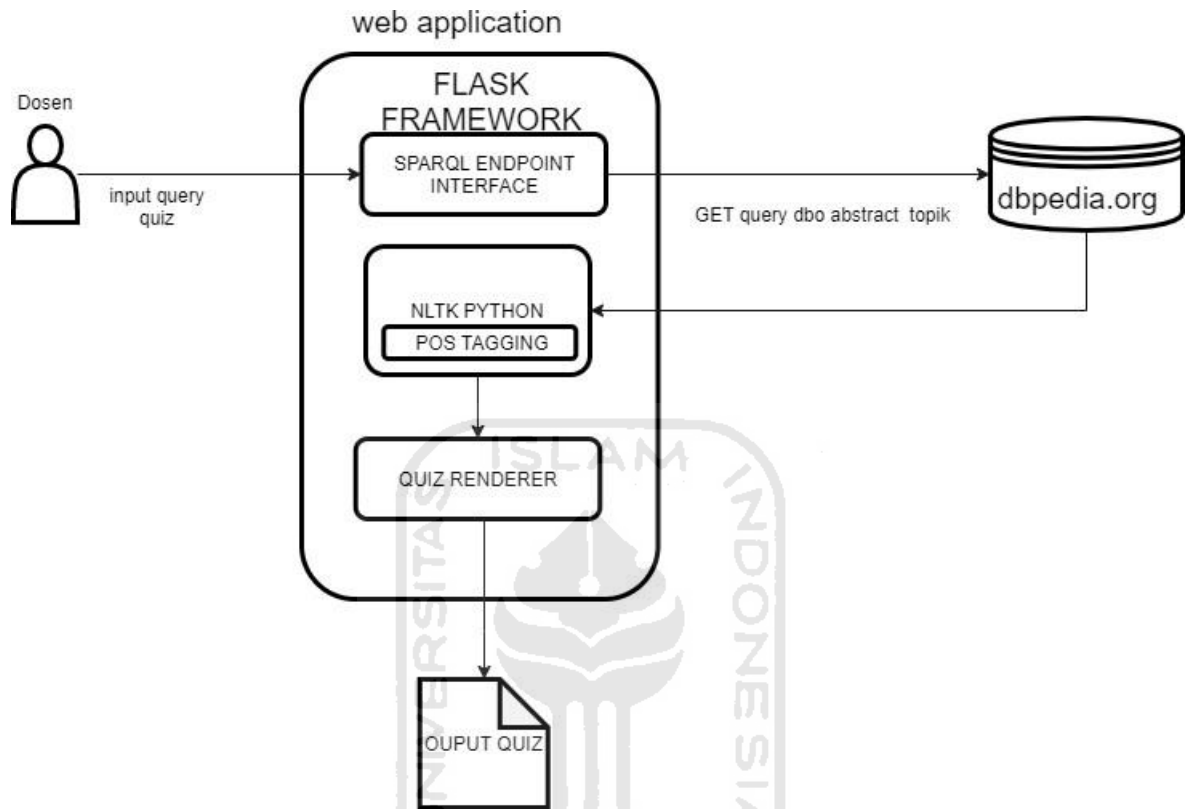


Gambar 3. 11 *Mockup* tampilan menambah kuis



3.7.4 Arsitektur back-end sistem

Pembuatan arsitektur *back-end* sistem bertujuan untuk memetakan gambaran proses yang terjadi secara umum sistem yang menggambarkan bagaimana konsep *linked data* dan pemanfaatan NLTK akan diimplementasikan dalam pembuatan sistem ini.



Gambar 3. 12 Desain arsitektur back-end sitem

Berdasarkan gambar 3.12 diatas menjelaskan bahwa sistem dapat menerjemahkan *query* yang diberikan kepada dosen dan langsung dapat diakses melalui *SPARQL endpoint* untuk mendapatkan data ekstraksi *dbo:abstract* dengan tipe data string yang mengandung rangkuman singkat informasi topik yang telah diinput yang telah dirangkum dalam *DBpedia.org*. Dengan method *GET*, data informasi dapat diambil dan dengan bantuan NLTK yang akan melakukan *POS tagging* pada tipe data string yang telah didapatkan untuk melakukan *rendering* kuis dengan melakukan penghilangan kata yang mengandung angka dan kuis dapat dihasilkan dengan mengisi bagian yang kosong pada teks yang ditampilkan pada tampilan *interface* sistem.

38 Rancangan Pengujian

Tujuan pembuatan rancangan pengujian sistem ini adalah untuk mengetahui adanya kesesuaian fungsionalitas sistem yang akan diimplementasikan dengan hasil analisis kebutuhan yang sudah dibuat.

3.8.1 Pengujian *Black Box Testing*

Pada sistem ini, pengujian dilakukan menggunakan metode *Black Box Testing*. Metode ini dilakukan untuk mengetahui bahwa sistem yang dibuat telah sesuai dengan persyaratan dan hasil pengujian metode ini berfokus pada hasil fungsi dan kegunaan sistem daripada aspek teknis. Skenario pengujian *Black Box* dapat dilihat pada Tabel 3.4 yang menunjukkan tahapan dan *test case* yang akan diujikan sebagai berikut.

Tabel 3. 4 Skenario *Black Box Testing*

<i>Use Case ID</i>	<i>Test case</i>	Kriteria evaluasi hasil pengujian	Validitas
UC-1	Mengosongkan <i>username</i> dan <i>password</i> saat meng-klik tombol <i>login</i> .	Muncul pesan untuk mengisi form login yang kosong.	
UC-2	Mengisi kolom <i>username</i> dengan dosen tanpa mengisi kolom <i>password</i> .	Muncul pesan untuk mengisi form login yang kosong.	
UC-3	<i>Input</i> topik kuis.	Sistem dapat melakukan <i>query</i> pada SPARQL endpoint dan menghasilkan kuis sesuai <i>query</i> yang dimasukkan.	
UC-4	<i>Input</i> kode kuis.	Sistem dapat menampilkan kuis hasil prose <i>query</i> dengan membagikan kode join yang akan	

<i>Use Case ID</i>	<i>Test case</i>	Kriteria evaluasi hasil pengujian	Validitas
		diinputkan oleh mahasiswa.	
UC-5	Manajemen kuis.	Sistem dapat menambah jumlah kuis dan menghapus kuis.	
UC-6	Mengerjakan soal dengan mengisi seluruh <i>radio button</i> .	Sistem akan menampilkan nilai akhir skor yang didapatkan.	
UC-7	Mengerjakan soal dengan mengisi sebagian seluruh <i>radio button</i> pilihan ganda. Kemudian kembali ke home.	Sistem akan menampilkan pilihan untuk kembali melanjutkan untuk menjawab pertanyaan lagi.	
UC-8	Manajemen mahasiswa.	Sistem dapat menambah dan menghapus hak akses mahasiswa untuk mengerjakan kuis.	
UC-9	Lihat nilai.	Sistem dapat menampilkan nilai hasil pengerjaan kuis yang bisa dilihat pada laman mahasiswa dan dosen.	

3.8.2 Pengujian User Acceptance Test

Pengujian *User Acceptance Test* yang dilakukan yang dilakukan bertujuan untuk melakukan untuk mengukur langsung kemudahan yang dirasakan oleh Pengguna yaitu dosen pada sistem yang telah diimplementasikan. UAT tidak jauh beda dengan kusioner pada tahap awal pembuatan aplikasi dengan hasil output sebuah dokumen hasil uji yang dapat dijadikan bukti bahwa sistem yang sudah diterima atau belum dalam memenuhi kebutuhan telah dibuat. Untuk mengetahui tanggapan responden yang berjumlah 3 orang dosen, maka pada penelitian ini maka rancangan pertanyaan peengujian yang dapat dilihat pada tabel 3.5 dengan 3 kategori penilaian yaitu dari aspek desain, kemudahan, dan efisiensi yang dapat dilihat sebagai berikut.

A = Sangat : Mudah/Bagus/Sesuai/Jelas

B = Mudah/Bagus/Sesuai/Jelas

C = Kurang : Mudah/Bagus/Sesuai/Jelas

D = Sulit/Jelek/Tidak Sesuai/Tidak Jelas

E = Sangat: Sulit/Jelek/Tidak Sesuai/Tidak Jelas

Tabel 3. 5 Pertanyaan Pengujian UAT

Aspek yang dinilai	NO	PERTANYAAN	A	B	C	D	E
Desain	1	Apakah tampilan <i>user interface</i> pada sistem yang diimplementasikan menarik ?					
Kemudahan	2	Menu atau fitur sistem yang diimplementasikan mudah untuk dipahami?					
	3	proses manajemen kuis dapat dilakukan dengan baik pada sistem ini?					
Efisien	4	Proses dalam menghasilkan kuis dapat efisien dilakukan dengan sistem ini?					
	5	Apakah dengan sistem penghasil kuis ini dapat dijadikan media evaluasi pembelajaran?					

3.8.3 Alat Penguji

Pada tahap pengujian ini diperlukan laptop dan akses internet yang memadai untuk pengujian langsung pada program yang telah dibuat.

3.8.4 Lokasi

Lokasi pengujian bertempat di lingkungan Fakultas Teknik Industri FTI.



BAB IV PEMBAHASAN

41 Batasan Implementasi

Hasil akhir dari implementasi aplikasi ini adalah untuk menghasilkan kuis yang digunakan dalam pembelajaran di perkuliahan di lingkungan FTI UII. Berdasarkan identifikasi masalah dan rancangan sistem yang telah dibuat, maka aplikasi ini diimplementasikan dalam bentuk *website* dengan bahasa pemrograman *Python* berbasis *framework Flask*. Hasil kuis yang muncul adalah merupakan implementasi dari metode *linked data* yang digunakan menggunakan *SPARQL wrapper*.

42 Spesifikasi Sistem

Hasil tahap perancangan dan analisis kebutuhan fungsional dan non-fungsional, sistem ini memiliki spesifikasi yang dapat dilihat pada tabel 4.1 di bawah.

Tabel 4. 1 Spesifikasi Sistem

Aspek	Spesifikasi
Platform.	Web application.
Bahasa pemrograman.	Python.
Framework.	FLASK.
Library.	alembic==1.4.2 astroid==2.3.3 autopep8==1.5.1 bcrypt==3.1.7 beautifulsoup4==4.9.0 blinker==1.4 cffi==1.14.0 click==7.1.1 cryptography==2.9 Flask==1.1.2 Flask-Admin==1.5.5 Flask-Login==0.5.0 Flask-Mail==0.9.1

Aspek	Spesifikasi
	Flask-Migrate==2.5.3 Flask-Script==2.0.6 Flask-Seeder==1.1.1 Flask-SQLAlchemy==2.4.1 Flask-User==1.0.2.2 Flask-WTF==0.14.3 isodate==0.6.0 isort==4.3.21 itsdangerous==1.1.0 Jinja2==2.11.1 lazy-object-proxy==1.4.3 lxml==4.5.0 Mako==1.1.2 MarkupSafe==1.1.1 mccabe==0.6.1 nltk==3.2.2 passlib==1.7.2 pycodestyle==2.5.0 pycparser==2.20 pylint==2.4.4 pylint-flask==0.6 pylint-plugin-utils==0.6 pyparsing==2.4.7 python-dateutil==2.8.1 python-editor==1.0.4 pytz==2019.3 rdflib==4.2.2 requests==2.13.0 six==1.14.0 soupsieve==2.0 SPARQLWrapper==1.8.5 SQLAlchemy==1.3.15

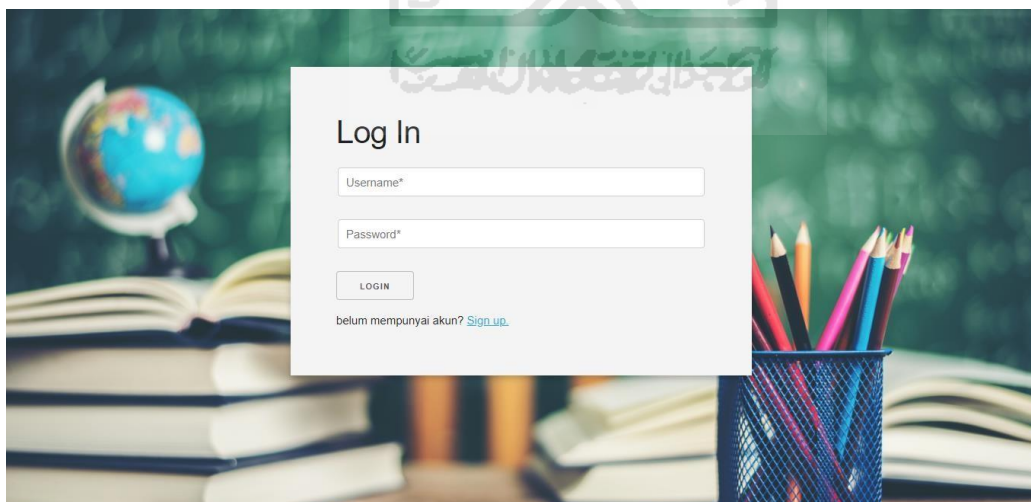
Aspek	Spesifikasi
	text2num==2.1.1 Werkzeug==1.0.1 wikipedia==1.4.0 wrapt==1.11.2 WTForms==2.2.1
<i>Sistem Operasi.</i>	<i>Windows.</i>

43 Implementasi Sistem

Berdasarkan metodologi penelitian yang dikembangkan dari tahap studi literatur, analisis kebutuhan, serta rancangan aplikasi. Berikut adalah hasil implementasi Sistem Penghasil Kuis Otomatis dengan Konsep *Linked Data*.

4.3.1 Halaman login

Gambar 4.1 merupakan antarmuka halaman *login* bagi dosen atau mahasiswa untuk dapat mengakses sistem menuju halamannya masing-masing dengan cara memasukkan *username* dan *password*.



Gambar 4. 1 Tampilan *Login*

4.3.2 Halaman *sign up*

Pada gambar 4.2 merupakan antarmuka halaman *sign up* bagi mahasiswa yang belum terdaftar pada sistem yang akan dibuat dengan mengisi *form* yang tersedia yaitu, *username*

dan *password* kemudian meng-klik tombol *sign up* dan kembali ke halaman login. Mahasiswa yang sudah mendaftar pada *form sign up* bisa masuk ke dalam sistem dengan mengisi *form login* yang kosong sesuai isian yang dimasukkan pada *form sign up*.

The image shows a 'Sign Up' form with the following fields and elements:

- First Name*
- Last Name*
- Username*
- Password*
- Confirm Password*
- * wajib diisi
- LOGIN button
- Sudah punya akun? [Log in](#)

Gambar 4. 2 Tampilan *sign up*

4.3.3 Halaman *home dashboard* mahasiswa

Pada Gambar 4.3 menunjukkan tampilan antarmuka *home* dari halaman mahasiswa. Disini kita dapat melihat kuis beserta topik yang sudah di upload oleh dosen melalui sistem yang ada. Terdapat beranda yang berisi informasi kuis yang sedang dikerjakan, jumlah soal kuis, tanggal dan waktu mahasiswa mulai mengerjakan kuis, dan batas akhir pengumpulan kuis.

The dashboard shows the following data:

2 Kuis yang masuk

0 kuis yang sudah dikerjakan

Kuis yang dikerjakan [+ masuk ke kuis](#)

NAMA KUIS	TOPIK	TANGGAL DIKERJAKAN	DEADLINE	SUMMARY	NILAI
kuis tengah semester	Asia	July 01, 2020 14:14:45 WIB	July 13, 2020 16:59:59 WIB	Remaining attempt(s) 3 attempt(s)	50

hasil

TOPIK	TANGGAL DIKERJAKAN	DEADLINE	NILAI AKHIR
Malaysia	2020-07-01 14:17:12.141298	2020-06-30 16:59:59	0

Gambar 4. 3 Tampilan *home dashboard* mahasiswa

4.3.4 Halaman histori nilai kuis mahasiswa

Pada gambar 4.4 adalah tampilan beranda untuk melihat histori kuis yang sudah pernah dikerjakan sebelumnya dan rincian nilai yang didapatkan.

NAMA KUIS	TOPIK	TANGGAL DIKERJAKAN	DEADLINE	SUMMARY	ACTION
Quiz Class 5	Topic 5	April 14, 2020 04:49:19 WIB	March 30, 2020 16:59:59 WIB	Final Highscore 0	
negara malaysia	Malaysia	June 30, 2020 08:01:36 WIB	June 30, 2020 16:59:59 WIB	Final Highscore 50	

Gambar 4. 4 Tampilan histori nilai kuis mahasiswa

4.3.5 Halaman masuk kuis

Pada gambar 4.5 adalah tampilan untuk masuk mengerjakan kuis yang telah dibuat oleh dosen. Mahasiswa hanya perlu memasukkan kode kuis dibagikan oleh dosen sebelumnya dan kuis bisa dikerjakan.

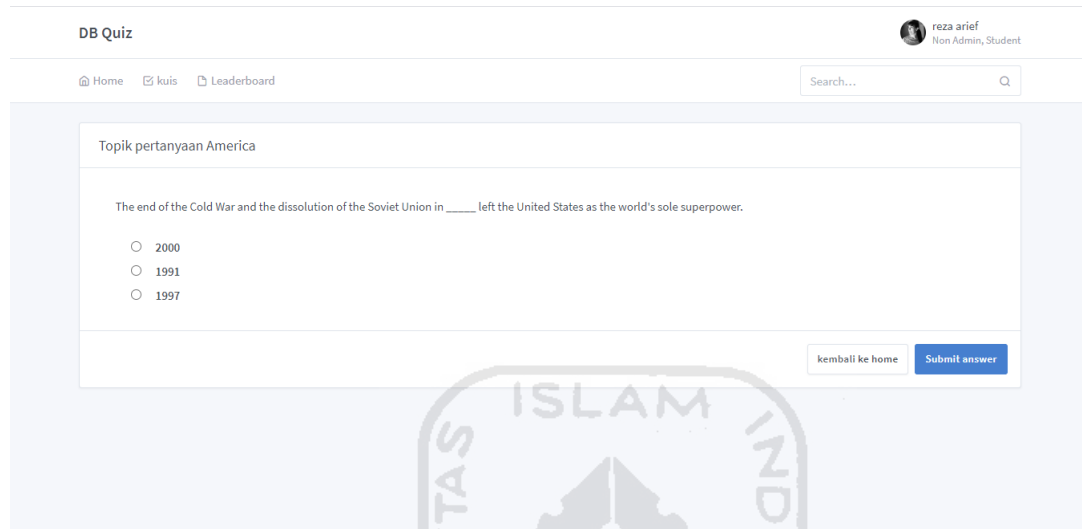
NAMA KUIS	TOPIK	TANGGAL DIKERJAKAN	DEADLINE	SUMMARY	NILAI
kuis tengah semester	Asia	July 01, 2020 14:14:45 WIB	July 13, 2020 16:59:59 WIB	Remaining attempt(s) 3 attempt(s)	50

TOPIK	TANGGAL DIKERJAKAN	DEADLINE	NILAI AKHIR
Malaysia	2020-07-01 14:17:12.141298	2020-06-30 16:59:59	0

Gambar 4. 5 Tampilan *input* kode kuis

4.3.6 Halaman kuis

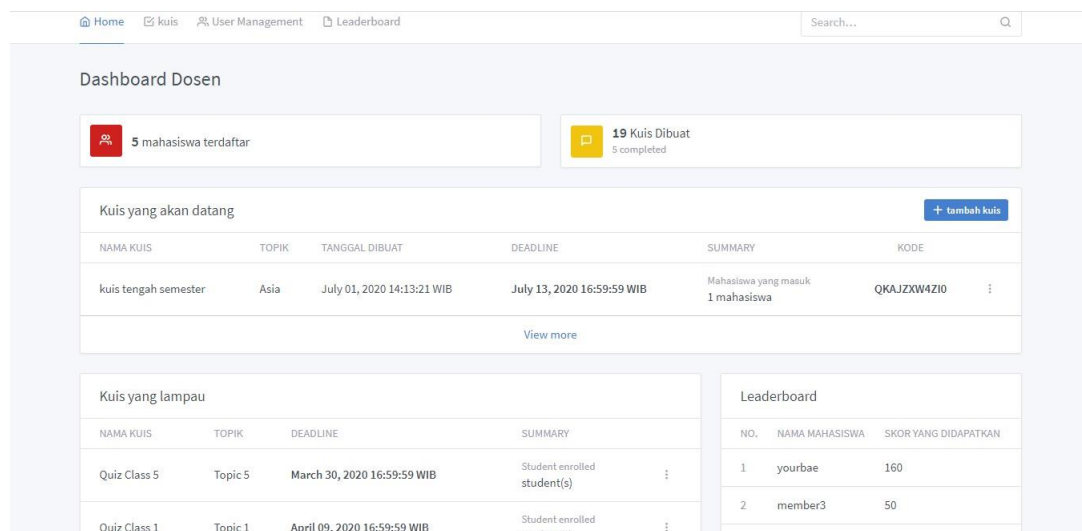
Pada gambar 4.6 menunjukkan bagaimana kuis yang ditampilkan pada sistem berbentuk pilihan ganda. Untuk pengerjaan kuis, mahasiswa cukup dengan meng-klik *radio button* dengan keterangan jawaban yang dirasa benar oleh mahasiswa.



Gambar 4. 6 Tampilan penyajian kuis

4.3.7 Halaman *home dashboard* dosen

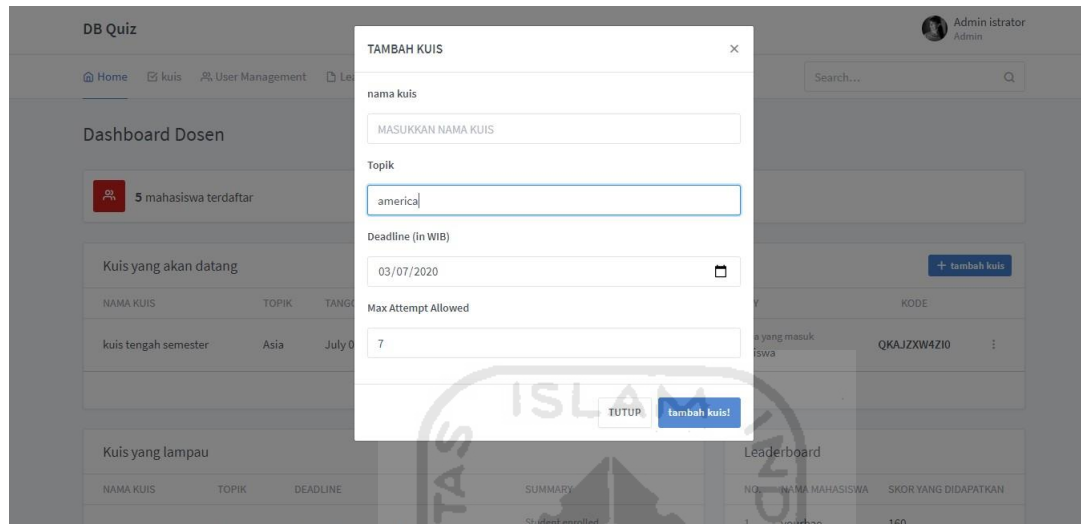
Pada gambar 4.7 menampilkan tampilan *home dashboard* dosen yang mempunyai segala hak akses pada sistem yang akan dibuat. Disini dosen dapat melihat kuis beserta topik yang sudah di input oleh dosen melalui sistem yang ada.



Gambar 4. 7 Tampilan *home dashboard* dosen

4.3.8 Halaman tambah kuis

Gambar 4.8 adalah tampilan untuk menambah kuis serta topik yang diinginkan pada sistem dengan melakukan *query* pencarian terhadap topik kuis yang diinginkan. Pada halaman ini juga mengatur waktu *deadline* pengumpulan kuis dan jumlah soal yang akan ditampilkan.



Gambar 4. 8 Tampilan menambah kuis

4.3.9 Halaman manajemen kuis

Gambar 4.9 adalah tampilan untuk manajemen kuis yang kuis serta topik yang diinginkan. Pada halaman ini dosen juga dapat mengetahui jawaban dari kuis yang telah dibuat oleh sistem serta menghapus kuis yang diinginkan seperti pada gambar 4.10.

DB Quiz Admin Istrator Admin

Home | **kuis** | User Management | Leaderboard Search...

Manajemen Kuis + tambah kuis

Show entries Search:

NAMA KUIS	TOPIK	KODE MASUK	TANGGAL DIBUAT	DEADLINE	STATUS	ACTION
Class 1	Google	QCGJZXW4Z19	May 03, 2020 13:23:20 WIB	May 03, 2020 16:59:59 WIB	Expired	Show report and questions
Kelas 7B	Microsoft Office	QKMJZXW4Z14	May 01, 2020 10:21:55 WIB	May 01, 2020 16:59:59 WIB	Expired	Show report and questions
Kelas 7H	Honda	QKHJZXW4Z19	April 14, 2020 06:26:42 WIB	April 14, 2020 16:59:59 WIB	Expired	Show report and questions
Kelas 7H	Google	QKGJZXW4Z18	May 03, 2020 11:25:23 WIB	May 03, 2020 16:59:59 WIB	Expired	Show report and questions
Kelas 7H	Google	QKGJZXW4Z11	May 03, 2020 11:26:55 WIB	May 03, 2020 16:59:59 WIB	Expired	Show report and questions
Kelas 7H	Google	QKGJZXW4Z13	May 03, 2020 11:29:09 WIB	May 03, 2020 16:59:59 WIB	Expired	Show report and questions

Gambar 4. 9 Tampilan manajemen kuis

Home | **kuis** | User Management | Leaderboard Search...

Class 1
nama kuis

May 03, 2020 16:59:59 WIB
Deadline

Google
Topik

QCGJZXW4Z19
masukkan kode

Topik: Google

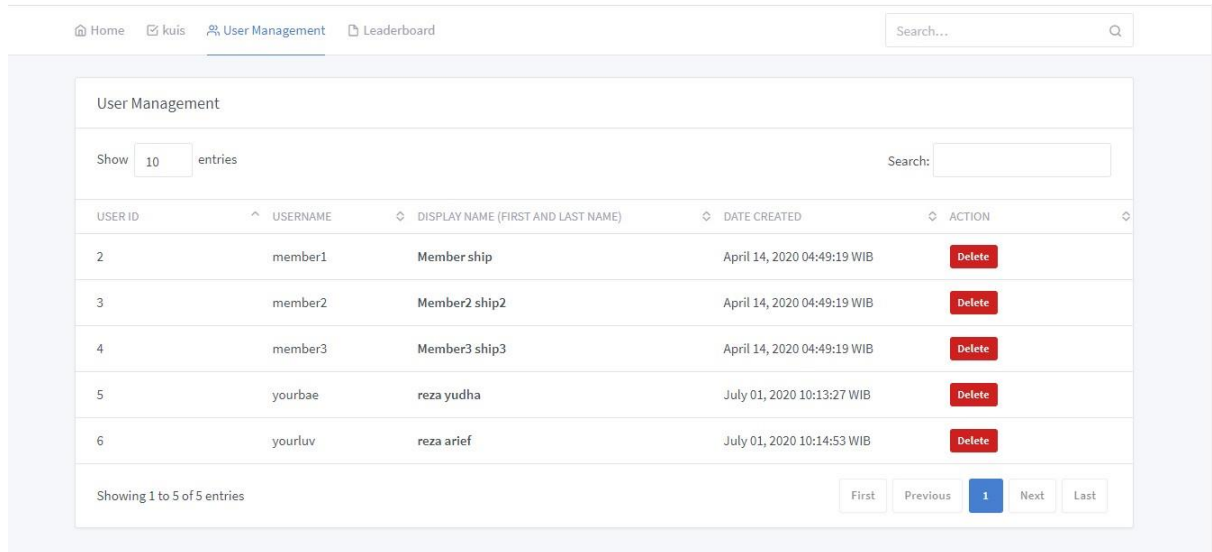
NO.	PERTANYAAN	JAWABAN
1	When this restructuring took place on October 2, _____, Google became Alphabet's leading subsidiary, as well as the parent for Google's Internet interests. Options: 2017, 2009, 2015	2015
2	In _____, a fiber-optic infrastructure was installed in Kansas City to facilitate a Google Fiber broadband service. Options: 2017, 2012, 2002	2012
3	In August _____, Google announced plans to reorganize its interests as a holding company called Alphabet Inc. Options: 2015, 2009, 2016	2015

hapus

Gambar 4. 10 Tampilan menghapus kuis

4.3.10 Halaman *user management* dosen

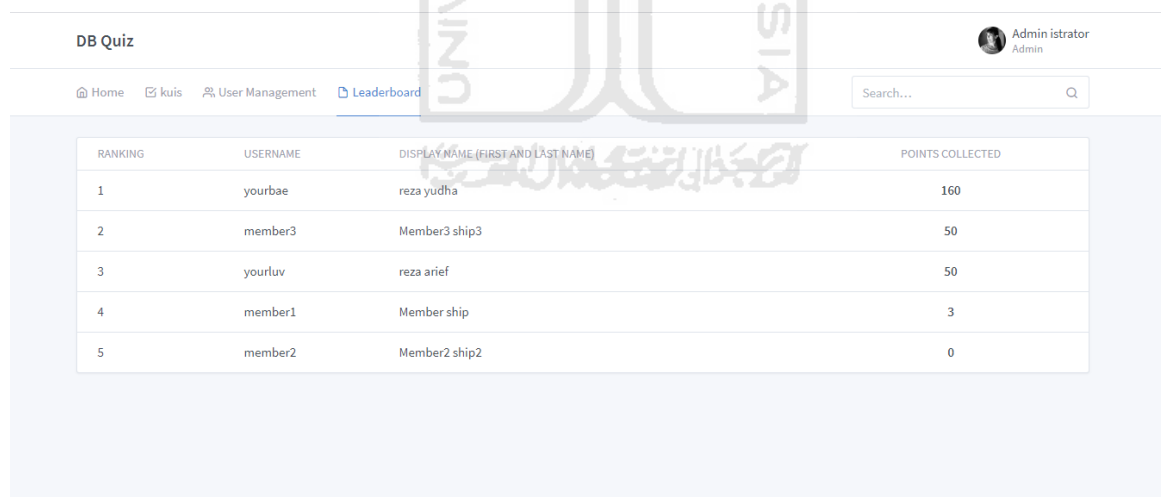
Pada gambar 4.1 adalah tampilan *user management* untuk mengurangi hak akses mahasiswa yang boleh masuk ke sistem dan mengerjakan kuis.



Gambar 4. 11 Tampilan *user management*

4.3.11 Halaman *leaderboard*

Pada gambar 4.12 menampilkan tabel *leaderboard* untuk mengetahui nilai terendah dan tertinggi dari mahasiswa yang telah mengerjakan kuis. Tampilan *leaderboard* juga bisa dilihat pada *dashboard* mahasiswa.



Gambar 4. 12 Tampilan *leaderboard*

4.3.12 Implementasi back-end

Berikut adalah baris kode dalam sistem ini dalam melakukan proses *query* dari DBpedia.

routes.py

```

from flask import Blueprint, render_template, redirect, url_for, request,
current_app, jsonify, make_response
from flask_user import roles_required
from ..models import User, Quiz, db, Role, Question
from datetime import datetime, timedelta
from .admin_forms import AddQuizForm, DeleteUserForm, DeleteQuizForm
from ..quiz_generator.article import Article
import pytz
from requests.exceptions import ConnectionError as ce
from ..helper.date_time_utils import local_time

admin_bp = Blueprint('admin_bp', _name__,
                    template_folder='templates', static_folder='static')

@admin_bp.route('/admin', methods=['GET'])
@roles_required('admin')
def root():
    return redirect(url_for('admin_bp.dashboard'))

@admin_bp.route('/admin/dashboard', methods=['GET'])
@roles_required('admin')
def dashboard():

    MAX_ENTRY_UPCOMING = 10
    MAX_ENTRY_LEADERBOARD = 5
    MAX_ENTRY_COMPLETED = 5

    data = dict()
    data['ct_registered_user'] = User.query.filter(User.roles ==
None).count()
    data['ct_quizzes_created'] = Quiz.query.count()
    data['completed_quizzes'] = Quiz.get_completed_quizzes()[
:MAX_ENTRY_COMPLETED]
    data['upcoming_quizzes'] =
Quiz.get_upcoming_quizzes()[ :MAX_ENTRY_UPCOMING]
    data['ct_quizzes_completed'] = len(data['completed_quizzes'])
    for i in range(len(data['upcoming_quizzes'])):
        data['upcoming_quizzes'][i].users_enrolled =
User.query.with_parent(
        data['upcoming_quizzes'][i]).count()
    for i in range(len(data['completed_quizzes'])):
        data['completed_quizzes'][i].users_completed = len(
User.get_student_completed_quiz_list(data['completed_quizzes'][i]))

    data['leaderboard'] = User.get_leaderboard()[ :MAX_ENTRY_LEADERBOARD]

    data['add_quiz_form'] = AddQuizForm()

    return render_template('index_admin.html', **data)

```

Ketika user melakukan konfirmasi dengan menekan tombol "add quiz", maka *request* tersebut akan di-*handle* oleh *method add_quiz*. *Request* yang diterima oleh *method* tersebut adalah *request* jenis *post*. Parameter yang dikirimkan yaitu *name*, *topic*, *deadline (date)*, *lang*.

Selanjutnya proses pembuatan sebuah pertanyaan akan ditangani oleh object dari class *Article.py*. Proses instansiasi object *Article* akan melakukan proses pengambilan data abstrak dengan bantuan *SPARQLWrapper* sebagai acuan dalam pengambilan data abstrak yaitu parameter '*topic*'.

Article.py

```
import nltk
from nltk.tokenize import sent_tokenize
import re
import urllib.request as urllib
import json

import requests
from bs4 import BeautifulSoup

import text_to_num as t2n
from .quiz import Quiz
from .question_sentence import QuestionSentence
from math import floor
from random import random, shuffle
from SPARQLWrapper import SPARQLWrapper, JSON

def dbpedia(q, lang):
    q = q.replace(' ', '_')

    if lang == 'en':
        sparql = SPARQLWrapper("http://dbpedia.org/sparql")
        query_string = "select ?artikel" + \
            " where { " + \
            f" dbr:{q}" + " dbo:abstract ?artikel.}"
    else:
        sparql = SPARQLWrapper(f"http://{lang}.dbpedia.org/sparql")

        query_string = "select ?artikel" + \
            " where { " + \
            f" dbpedia-id:{q}" + " dbpedia-owl:abstract ?artikel.}"

    sparql.setQuery(query_string);
    sparql.setReturnFormat(JSON)

    result = sparql.query().convert()['results']['bindings'];

    if len(result) == 0:
        return None

    try:
        text = result[0]['artikel']['value']
        return text
    except Exception as e:
        print(str(e))
        return None

class Article():

    def __init__(self, name, lang):
```

```

self.name = name

self.page = dbpedia(name, lang)
if self.page == None:
    return

self.quiz = Quiz([])

self.generate_questions_for(self.page.encode('utf8'))

def get_question_data(self, s):
    tokens = nltk.word_tokenize(s)
    tagged = nltk.pos_tag(tokens)
    grammar = """
        NUMBER: {<$>*<CD>+<NN>*}
        LOCATION: {<IN><NNP>+<, | IN><NNP>+}
        PROPER: {<NNP|NNPS><NNP|NNPS>+}
        """

    #
    # HIT!: {<PROPER><NN>?<VBZ|VBN>+}
    # DATE: {<IN>(<$>*<CD>+<NN>*)}

    chunker = nltk.RegexpParser(grammar)
    result = chunker.parse(tagged)
    return result

def generate_questions_for(self, sec):
    _sec = "".join(re.split('\(',
        sec.decode("utf-8").replace(")",
"(")) [0::2])

    for sentence in sent_tokenize(_sec):
        qdata = self.get_question_data(sentence)
        if len(qdata) >= 75 and len(qdata) <= 150:
            qdata = []

            self.create_questions(sentence, qdata)

def create_questions(self, sentence, chunked):
    gaps = []

    for word in chunked:
        if type(word) != tuple:
            target = []
            for y in word:
                target.append(y[0])
            orig_phrase = " ".join(target)

            if word.label() == "NUMBER":
                modified_phrase = orig_phrase[:]

            try:

                modified_phrase = t2n.text2num(
                    modified_phrase, lang=T2N_LANG)
            except:

```

```

        try:
            test = int(modified_phrase) + \
                  float(modified_phrase)
        except:

            continue

        if self.probably_range(modified_phrase):
            return

        gaps.append((word.label(), orig_phrase,
modified_phrase))
        elif word.label() in ["LOCATION", "PROPER"]:
            gaps.append((word.label(), orig_phrase, orig_phrase))

    if len(gaps) >= 1 and len(gaps) == len(set(gaps)):

        gaps_filtered = [gap for gap in gaps if gap[0]
                        == 'NUMBER' or gap[0] == 'LOCATION']

        if len(gaps_filtered):
            self.quiz.add(QuestionSentence(sentence, gaps_filtered))

```

Kemudian pada pemanggilan proses `generate_questions_for`, terdapat proses untuk melakukan *cleaning word* dari berbagai simbol yang tidak dibutuhkan serta membagi data abstrak menjadi beberapa kalimat yang disimpan di array.

```

def generate_questions_for(self, sec):
    # Rid of all parentheses for easier processing
    _sec = "".join(re.split('\(|\)',
        sec.decode("utf-8").replace(")",
"(")) [0::2])

    for sentence in sent_tokenize(_sec):
        qdata = self.get_question_data(sentence)
        if len(qdata) >= 75 and len(qdata) <= 150:
            qdata = []

        self.create_questions(sentence, qdata)

```

Per kalimatnya akan didapatkan sebuah pertanyaan dengan memanggil method `get_question_data` dan `create_question`.

Article.py

```

def get_question_data(self, s):
    tokens = nltk.word_tokenize(s)
    tagged = nltk.pos_tag(tokens)
    grammar = ""
    NUMBER: {<$>*<CD>+<NN>*}

```



```

    chunker = nltk.RegexpParser(grammar)
    result = chunker.parse(tagged)
    return result

def create_questions(self, sentence, chunked):
    gaps = []
    for word in chunked:
        if type(word) != tuple:
            target = []
            for y in word:
                target.append(y[0])
            orig_phrase = " ".join(target)

            if word.label() == "NUMBER":
                modified_phrase = orig_phrase[:]

                try:
                    except:
                        try:
                            test = int(modified_phrase) + \
                                float(modified_phrase)
                        except:
                            if self.probably_range(modified_phrase):
                                return

                gaps.append((word.label(), orig_phrase,
modified_phrase))

                elif word.label() in ["LOCATION", "PROPER"]:
                    gaps.append((word.label(), orig_phrase, orig_phrase))

            if len(gaps) >= 1 and len(gaps) == len(set(gaps)):

                gaps_filtered = [gap for gap in gaps if gap[0]
                    == 'NUMBER' ]

                if len(gaps_filtered):
                    self.quiz.add(QuestionSentence(sentence, gaps_filtered))

class Quiz():

    def __init__(self, questions):

        self.questions = questions

    def add(self, question):
        self.questions.append(question)

    def get_ten_random(self):
        return _sample(self.questions, 10)

```

Pada *method get_question_data*, hal yang dilakukan pertama kali adalah melakukan tokenisasi. Setelah ditoken, setiap token kata akan diberikan pos tag. Kemudian, jika ada token kata yang membentuk struktur pos tag tertentu, maka gabungan token kata tersebut akan diidentifikasi sebagai Angka. Sebagai contoh: jika ada sebuah kombinasi token yang sesuai

dengan pola kata yang diawali dengan kata preposisi seperti 'ke','pada,'di' yang kata selanjutnya diikuti dengan angka, maka kombinasi token tersebut dikategorikan sebagai Tag '*NUMBER*'. Kalimat yang memiliki pola tersebut akan ditambahkan di array sementara. Kemudian dari semua kalimat yang telah ditambahkan diambil 10 pertanyaan random yang memiliki tag '*NUMBER*'.

Article.py

```
def get_question_set(text, gaps):
    max_g, min_g = len(gaps) - 1, 0
    gap_index = get_random_int(min_g, max_g)
    correct = gaps[gap_index]
    correct_answer = correct[2]
    label = correct[0]

    redacted_text, redacted_ans = convert_to_redacted(text, correct[1],
label)

    try:

        if label == 'LOCATION':
            wrong_answers = get_wrong_locations(
                correct_answer
            )
        elif label == 'NUMBER':
            wrong_answers = get_wrong_answers(
                label, correct_answer, gaps, gap_index)
        else:
            return None

        wrong_answers += [correct_answer]
        shuffle(wrong_answers)

        return {
            "question": redacted_text,
            "opt1": str(wrong_answers[0]),
            "opt2": str(wrong_answers[1]),
            "opt3": str(wrong_answers[2]),
            "answer": str(correct_answer)
        }
    except:
        return None
```

Proses pembuatan opsi jawaban dari pertanyaan dilakukan pada *method get_question_set*. Pada *method* ini, kata yang merupakan sebuah jawaban akan digantikan dengan tanda '_' terlebih dahulu. Kemudian dilakukan proses pembuatan 3 jawaban lain yang salah. Tag '*NUMBER*' akan menggunakan *method get_wrong_answer*.

Article.py

```
def get_wrong_answers(label, correct_answer, other_gaps, gap_index):
    MAX_RETRIES = 20
    wrong_1 = str(get_wrong_answer(
```

```

    label, correct_answer, other_gaps, gap_index))
wrong_2 = str(get_wrong_answer(
    label, correct_answer, other_gaps, gap_index))
any_answer_same = None
for i in range(MAX_RETRIES):
    any_answer_same = (wrong_1 == wrong_2) or (
        wrong_1 == correct_answer) or (wrong_2 == correct_answer)

    if not any_answer_same:
        break

    wrong_1 = str(get_wrong_answer(
        label, correct_answer, other_gaps, gap_index))
    wrong_2 = str(get_wrong_answer(
        label, correct_answer, other_gaps, gap_index))

if (any_answer_same):
    to_ret = [str(int(wrong_1) + 1), str(int(wrong_2) - 1)]
    return to_ret

return [wrong_1, wrong_2]

```

Pada *method get_wrong_answer*, untuk mendapatkan angka jawaban yang salah dilakukan dengan mengambil sampel nilai random dengan rentang antara (angka - 50) hingga angka (angka + 30). jika sampel ternyata kebetulan sama dengan jawaban salah sebelumnya atau sama dengan jawaban yang benar. maka proses tersebut akan diulangi sebanyak 20 kali hingga mendapatkan 3 jawaban salah yang berbeda.

44 Pengujian Sistem

Sesuai dengan skenario rancangan pengujian yang telah dibuat, dengan menggunakan metode *Black Box Testing* untuk mengetahui bahwa sistem yang dibuat telah sesuai dengan persyaratan serta fungsi dan kegunaan sistem daripada aspek teknis. Hasil dari skenario pengujian *Black Box* dapat dilihat pada tabel 4.2 berikut.

Tabel 4. 2 Hasil pengujian *Black Box*

<i>Use Case ID</i>	<i>Test case</i>	Kriteria evaluasi hasil pengujian	Validitas	Keterangan
UC-1	Mengosongkan <i>username</i> dan <i>password</i> saat mengklik tombol <i>login</i> .	Muncul pesan untuk mengisi form login yang kosong.	Valid	Dapat dilihat di Lampiran IV.
UC-2	Mengisi kolom <i>username</i> dengan dosen tanpa mengisi kolom <i>password</i> .	Muncul pesan untuk mengisi form login yang kosong.	Valid	Dapat dilihat di Lampiran IV.
UC-3	<i>Input</i> topik kuis.	Sistem dapat melakukan <i>query</i> pada SPARQL endpoint dan menghasilkan kuis sesuai <i>query</i> yang dimasukkan.	Valid	Dapat dilihat di Lampiran IV.
UC-4	<i>Input</i> kode kuis.	Sistem dapat menampilkan kuis hasil prose <i>query</i> dengan membagikan kode join yang akan diinputkan oleh mahasiswa.	Valid	Dapat dilihat di Lampiran IV.
UC-5	Manajemen kuis.	Sistem dapat mengurangi topik kuis yang ada.	Valid	Dapat dilihat di Lampiran IV.
UC-6	Mengerjakan soal dengan mengisi seluruh <i>radio button</i> .	Sistem akan menampilkan nilai	Valid	Dapat dilihat di Lampiran IV.

<i>Use Case ID</i>	<i>Test case</i>	Kriteria evaluasi hasil pengujian	Validitas	Keterangan
		akhir skor yang didapatkan.		
UC-7	Mengerjakan soal dengan mengisi sebagian seluruh <i>radio button</i> pilihan ganda.	Sistem akan menampilkan pilihan untuk kembali melanjutkan untuk menjawab pertanyaan lagi.	Valid	Dapat dilihat di Lampiran A IV.
UC-8	Manajemen mahasiswa.	Sistem dapat menghapus hak akses mahasiswa untuk mengerjakan kuis.	Valid	Dapat dilihat di Lampiran A IV.
UC-9	Manajemen nilai.	Sistem dapat menampilkan nilai hasil pengerjaan kuis yang bisa dilihat pada laman mahasiswa dan dosen.	Valid	Dapat dilihat di Lampiran A IV.

Hasil dari pengujian *user acceptance test* yang telah dirancang sebelumnya yang digunakan sebagai bukti bahwa sistem yang telah diimplementasikan dapat diterima atau tidaknya oleh pengguna dan memenuhi kebutuhan yang telah dibuat dengan mengetahui tanggapan responden yang berjumlah 3 orang dosen dapat dilihat pada tabel 4.3 dan kemudian datanya diolah dan dikelompokkan berdasarkan 3 aspek penilaian yang dan datanya disajikan dalam bentuk persentase pada tabel 4.4 dibawah sebagai berikut.

Tabel 4. 3 Hasil pengujian UAT

Aspek yang dinilai	NO	PERTANYAAN	A	B	C	D	E
Desain	1	Apakah tampilan <i>user interface</i> pada sistem yang diimplementasikan menarik ?		2		1	
Kemudahan	2	Menu atau fitur sistem yang diimplementasikan mudah untuk dipahami?			3		
	3	proses manajemen kuis dapat dilakukan dengan baik pada sistem ini?			2	1	
Efisien	4	Proses dalam menghasilkan kuis dapat efisien dilakukan dengan sistem ini?		2		1	
	5	Apakah dengan sistem penghasil kuis ini dapat dijadikan media evaluasi pembelajaran?		2	1		

Tabel 4. 4 Persentase hasil pengujian UAT

Aspek yang dinilai	Pertanyaan	Persentase (%)				
		A	B	C	D	E
Desain	P1		66,6%		33,4%	
Persentase kumulatif(%)			66,6%		33,4%	
Kemudahan	P2			100%		
	P3			66,6%	33,4%	
Persentase kumulatif(%)				83,3%	16,7%	
Efisien	P4		66,6%		33,4%	
	P5		66,6%	33,4%		
Persentase kumulatif(%)			66,6%	16,7%	16,7%	

A = Sangat : Mudah/Bagus/Sesuai/Jelas

B = Mudah/Bagus/Sesuai/Jelas

C = Kurang : Mudah/Bagus/Sesuai/Jelas

D = Sulit/Tidak bagus/Tidak Sesuai/Tidak Jelas

E = Sangat: Sulit/Tidak bagus/Tidak Sesuai/Tidak Jelas

45 Hasil Pengujian sistem

Berdasarkan hasil pengujian sistem yang telah dilakukan dengan pengujian *black box* dan pengujian *user acceptance*, maka dapat diambil kesimpulan bahwa dalam pengujian ini.

1. Berdasarkan pengujian *black box* sudah sesuai dengan rancangan pengujian yang telah dibuat.
2. Berdasarkan pengujian *user acceptance* dengan koresponden 3 dosen dan 3 aspek penilaian, yaitu desain (tampilan), kemudahan dan efisien. Dari ketiga aspek tersebut dapat disimpulkan bahwa sebanyak 66,6 % koresponden menyatakan bagus dan menarik terhadap desain tampilan sistem, sebanyak 33,4% menyatakan kurang bagus terhadap tampilan sistem . Kemudian, dari aspek kemudahan sebanyak 83,3 % koresponden menyatakan bahwa masih kurang mudah dan sesuai terhadap implementasi fitur yang ditawarkan oleh sistem dan 16,7 % koresponden menyatakan bahwa kesusahan terhadap penggunaan fitur yang ditawarkan oleh sistem. Kemudian, dari aspek efisiensi sebanyak 66,6% koresponden menyatakan bahwa sistem yang ditawarkan dapat digunakan sebagai media yang efisien dalam menyajikan kuis, sebanyak 16,7 % koresponden masih menyatakan kurang tidak sesuai sebagai media yang efisien dalam menyajikan kuis, dan 16,7% koresponden menyatakan masih tidak sesuai sebagai media yang efisien dalam menyajikan kuis.

46 Keterbatasan Sistem

Berdasarkan hasil pengujian diatas juga didapatkan keterbatasan yang belum bisa untuk diimplementasikan ke dalam yang dapat diuraikan sebagai berikut.

1. Input query topik yang diinginkan ntidak ditemukan pada data yang telah terangkum pada DBpedia. Hal ini dikarenakan tidak semua artikel yang terdapat pada DBpedia mencakup segala pengetahuan yang dibutuhkan dan pada kasus lainnya pada beberapa kasus, proses pembuatan kuis tidak berhasil karena tidak

terdeteksinya *tag* 'NUMBER' pada data string yang sudah dirangkum dalam *dbo:abstract* pada *DBpedia.org* yang didapatkan dari *SPARQLWrapper*.

2. Sistem manajemen kuis yang dirasa kurang maksimal karena belum bisa melakukan manajemen kuis per topik kuis yang dihasilkan. Sistem yang telah diimplementasikan hanya bisa melakukan manajemen kuis dan nilai yang didapat mahasiswa berdasarkan kumulatif topik kuis apa saja yang telah dikerjakan oleh mahasiswa.
3. Jumlah pertanyaan kuis yang dihasilkan belum dapat diatur. Hal ini karena bergantung jumlah kata dalam paragraf atau kalimat yang mengandung *POS Tag* 'NUMBER' yang terkandung dalam data string pada *dbo:abstract* yang berasal dari *DBpedia.org*.



BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian dan pengujian yang telah dijelaskan pada bab sebelumnya, Sistem Penghasil Kuis Otomatis Berdasarkan *Linked Data*, dapat disimpulkan bahwa :

1. Pemanfaatan metode *Linked Data* dan Dbpedia sebagai sumber materi pada Sistem Penghasil Kuis Otomatis Berdasarkan *Linked Data* berbasis web berhasil diimplementasikan dalam menghasilkan kuis sesuai topik yang diinginkan.
2. Sistem penghasil kuis otomatis berdasarkan *Linked Data* dapat berfungsi sesuai dengan kebutuhan fungsionalitasnya dan dapat digunakan sebagai media yang efisien dalam menyajikan kuis, namun masih ada kendala dalam aspek kemudahan pengguna yang dirasa masih kurang intuitif berdasarkan hasil pengujian yang telah dilakukan.

5.2 Saran

Saran peneliti untuk pengembangan sistem selanjutnya sebagai berikut:

1. Pada penelitian ini sistem yang dibuat dengan platform *website*. pada penelitian selanjutnya peneliti berharap dapat mengembangkan ke dalam platform lain seperti aplikasi android.
2. Pada penelitian ini sistem yang telah diimplementasikan hanya bisa menghasilkan pertanyaan dengan basis jawabannya hanya sekadar angka dan berbentuk pilihan ganda. Pada penelitian selanjutnya diharapkan sistem dapat menghasilkan kuis yang basis jawabannya tidak hanya angka, namun pertanyaan yang bisa mengandung unsur kalimat pertanyaan apa, siapa, dimana, bagaimana, kapan, dan juga bisa mengolah pertanyaan dari data ekstraksi *dbo:abstract* yang mengandung unsur 'lokasi' atau 'tempat' dan 'orang' dan berbentuk isian singkat agar variasi pertanyaan yang dihasilkan bisa beragam.
3. Memaksimalkan fitur untuk mengatur jumlah pertanyaan yang dihasilkan. Sistem yang sudah diimplementasikan sebelumnya sudah ada, namun belum bisa berfungsi.

4. Menambah fitur manajemen kuis yang ada agar bisa melakukan manajemen kuis dan perhitungan nilai bagi setiap mahasiswa yang mengerjakan kuis per topik kuis yang dikerjakan.
5. Pada penelitian ini hasil implementasi sistem hanya diuji dengan metode *Black Box Testing* dan *User Acceptance Test* dengan koresponden dosen. Pada penelitian selanjutnya peneliti bisa melakukan pengujian dengan metode *Black Box Testing* dan *User Acceptance Test* dengan koresponden mahasiswa untuk mengukur langsung kemudahan yang dirasakan oleh mahasiswa.
6. Memperbagus tampilan user interface dan juga membuatnya lebih intuitif yang telah ada dengan mencontoh aplikasi kuis yang sudah ada.



DAFTAR PUSTAKA

- Anggadini, S. D. (2013). Analisis Sistem Informasi Manajemen Berbasis Komputer dalam Proses Pengambilan Keputusan. *Majalah Ilmiah Unikom*.
- Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., & Ives, Z. (2007). DBpedia: A nucleus for a Web of open data. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. https://doi.org/10.1007/978-3-540-76298-0_52
- Barbieri, D. F., Braga, D., Ceri, S., Della Valle, E., & Grossniklaus, M. (2009). C-SPARQL: SPARQL for continuous querying. *WWW'09 - Proceedings of the 18th International World Wide Web Conference*. <https://doi.org/10.1145/1526709.1526856>
- Bird, S., Klein, E., & Loper, E. (2009). Language Processing and Python. *Computing*.
- Bizer, C., Heath, T., & Berners-Lee, T. (2009). Linked data - The story so far. *International Journal on Semantic Web and Information Systems*. <https://doi.org/10.4018/jswis.2009081901>
- Dr. Kusnendi, M. S. (2014). Konsep Dasar Sistem Informasi. *Konsep Dasar Sistem Informasi*.
- Dwiono, A. (2013). Mesin Pencari Cerdas Dengan Web Semantik. *Jurnal Generic*.
- Grinberg, M. (2014). Flask Web Development. *O'Reilly*. <https://doi.org/10.1007/s13398-014-0173-7.2>
- Heath, T., & Bizer, C. (2011). Linked Data: Evolving the Web into a Global Data Space. *Synthesis Lectures on the Semantic Web: Theory and Technology*. <https://doi.org/10.2200/S00334ED1V01Y201102WBE001>
- Ibrahim, N. (2007). Pengembangan Aplikasi Semantic Web Untuk Membangun Web yang Lebih Cerdas. *Jurnal Informatika*.
- Klyne, G., & Carroll, J. J. (2004). Resource Description Framework (RDF): Concepts and Abstract Syntax. W3C Recommendation 10 February 2004. In W3C.
- Liu, D., & Lin, C. (2014). Sherlock: A semi-automatic quiz generation system using linked data. *CEUR Workshop Proceedings*.
- Miao, Q., Fang, R., & Meng, Y. (2015). Extracting and integrating nutrition related linked data. *Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing, IEEE ICSC 2015*. <https://doi.org/10.1109/ICOSC.2015.7050835>
- Ngafifi, M. (2014). KEMAJUAN TEKNOLOGI DAN POLA HIDUP MANUSIA DALAM PERSPEKTIF SOSIAL BUDAYA. *Jurnal Pembangunan Pendidikan: Fondasi Dan*

Aplikasi. <https://doi.org/10.21831/jppfa.v2i1.2616>

Pambudi, C. P., & Sismoro, H. (2013). PEMBUATAN APLIKASI KUIS PERANGKAT KOMPUTER BERBASIS ANDROID. *Data Manajemen Dan Teknologi Informasi (DASI)*.

Tanwir, T., Rahman F, A., & Rahman F, A. (2018). DAMPAK PENGGUNAAN MEDIA PEMBELAJARAN BERBASIS ICT TERHADAP HASIL BELAJAR PAI PESERTA DIDIK PADA SMK NEGERI 1 KOTA PAREPARE. *AL-ISHLAH: Jurnal Pendidikan Islam*. <https://doi.org/10.35905/alishlah.v16i1.732>

Vega-Gorgojo, G. (2019). Clover quiz: A trivia game powered by dbpedia. *Semantic Web*. <https://doi.org/10.3233/SW-180326>

W3C. (2008). Cool URIs for the Semantic Web. *W3C Interest Group Note*.



LAMPIRAN

Transkrip wawancara

Biodata partisipan

Biodata wisnu cahyo prabowo

A. Identitas Diri

1.	Nama Lengkap (dengan Gelar)	Wisnu Cahyo Prabowo S. Farm., M. Si, Apt
2.	Jenis Kelamin	Laki-Laki
3.	NIP	0017068804
4.	NIDN	8814830017
5.	Tempat dan tanggal lahir	Balikpapan 17 Juni 1988
6.	Alamat E-mail	wisnu@farmasi.unmul.ac.id
7.	Nomor HP	0813 50088401
8.	Nama Institusi Tempat Kerja	Fakultas Farmasi Universitas Mulawarman
9.	Alamat Kantor	Kampus UNMUL Gunung Kelua, Gedung Farmasi, Jalan Penajam. Kota Samarinda Provinsi Kalimantan Timur
10.	Nomor Tel/Faks	0541-739491

B. Biodata Lainnya

Riwayat Pendidikan Terakhir

Perguruan Tinggi	Institut Teknologi Bandung
Jenjang Pendidikan	Magister (S2)
Kelompok Keilmuan	Farmasi
Tahun Lulus	2013

Transkrip wawancara

1. Sejak kapan Bapak berprofesi sebaga dosen?

Jawaban : Oktober 2013

2. Metode apa saja yang Bapak gunakan dalam mengevaluasi hasil mahasiswa?

Jawaban : Ujian tertulis, tugas, presentasi, diskusi

3. Apakah Bapak sering memberikan kuis kepada mahasiswa pada saat mengajar?

Jawaban :Iya. Sebagai tahapan pokok pembelajaran

4. Biasanya kuis diberikan pada saat sebelum kegiatan pengajaran dimulai atau sesudah?

Jawaban :Sesudah materi atau selesai termin / 1/2 semester

5. Biasanya kuis yang diberikan apakah secara lisan atau tulisan?

Jawaban :Bisa keduanya

6. Apakah Bapak pernah memberikan kuis secara daring/online?

Jawaban : Tidak pernah. Daring hanya uts/uas

7. Apakah Bapak mengetahui aplikasi pembuat kuis yang sering atau pernah digunakan pada sata pengajaran?

Jawaban :Iya tau

8. Apakah penggunaan aplikasi tersebut dirasa efektif dan efisien dalam menyajikan kuis kepada mahasiswa?

Jawaban :Iya efisien. Kurang efektif pada matakuliah tertentu. Seperti kajian kasus dan kemampuan verbal dan pola fikir cpt mahasiswa.

9. Bagaimana pendapat Bapak tentang aplikasi yang dapat menghasilkan kuis secara otomatis dengan mengetikkan keyword sesuai yang ibu inginkan?

Jawaban :Sangat bagus sekali, tinggal sosialisasi ke mahasiswa sebagai objeknya.

CURRICULUM VITAE

Biodata Partisipan



Nama : Risna Agustina, M.Si., Apt.

Tempat dan tanggal lahir : Samarinda, 29 Agustus 1988

Kebangsaan : Indonesian

Agama : Islam

Profesi : Dosen

Alamat Kerja : Fakultas Farmasi, Universitas
Mulawarman, Kampus UNMUL gn.
Kelua Samarinda, 75119, Indonesia

Office Phone : +62 (0541) 7952144

Office Fax : +62 (0541) 739491

Nomor Telpon :

Alamat E-Mail resmi : risna@farmasi.unmul.ac.id

Alamat E-Mail : aaisyahrisna@yahoo.com

Department : Farmasi

Education Background:

- Undergraduate, Pharmacy, Mulawarman University, Indonesia, September 2006 – April 2011

- Thesis: Study of The Pattern of Hypertension Treatment in Hospitalized Patients in the Hospital Abdul Wahab Sajranie
- Master Degree, Pharmacy, Bandung Institute of Technology, Indonesia, July 2011 – July 2013
 - Thesis: The Effectiveness of “ GI Acupuncture” to The Stress Treatment In Patients at Sukamenak Acupuncture Clinical and UPT Health Services Bumi Medika Ganesa ITB.
- Profession Degree (Apothecary/Pharmacist), Mulawarman University, Indonesia, September 2015 – November 2016

Participant and Speaker of National Seminar:

- 2nd Mulawarman Pharmaceutical Conferences, as participant and oral presenter at 24-25 October 2015
- 3rd Mulawarman Pharmaceutical Conferences, as participant and oral presenter at 20-21 April 2016
- 4th Mulawarman Pharmaceutical Conferences, as participant and oral presenter at 26-27 October 2016
- 5th Mulawarman Pharmaceutical Conferences, as participant and oral presenter at 23-24 April 2017
- 6th Mulawarman Pharmaceutical Conferences, as participant and oral presenter at 07-08 November 2017
- 7th Mulawarman Pharmaceutical Conferences, as participant and oral presenter at 12-13 May 2018

Employment History:

- Head of Clinical and Community Pharmacy Laboratory at Mulawarman University (Since 2014 until now)

Society Membership:

- Association of Indonesian Pharmacist Branch of East Borneo as member.

List of Publications (Journal and Proceeding):

- **Risna Agustina, Dewi Tita Indrawati, Muhammad Amir Masruhim.** ACTIVITIES OF SALAM LEAVES EXTRACT (*EUGENIA POLYANTHA*) AS ANTIINFLAMATION ON RATTUS NORVEGICUS. Journal of Tropical Pharmacy and Chemistry, 2015 Vol 3. Issue.2
- **Risna Agustina, Nurul Annisa, Wisnu Cahyo Prabowo.** POTENTIAL INTERACTION OF DRUG RECIPIENTS OF HYPERTENSION PATIENTS IN ONE OF THE GOVERNMENT HOSPITALS IN SAMARINDA CITY, Jurnal Sains dan Kesehatan, 2015 Vol 1. Issue. 4.
- **Risna Agustina, Andreanus Soemardji.** THE EFFECTIVENESS of “ GI ACUPUNTURE” to The STRESS TREATMENT In PATIENTS at SUKAMENAK ACUPUNTURE CLINICAL and UPT HEALTH SERVICES BUMI MEDIKA GANESA ITB. Jurnal Sains dan Kesehatan, 2016 Vol 1. Issue. 5

- **Risna Agustina, Wisnu Cahyo Prabowo.** STUDY Of INSULIN REGIMEN IN TYPE II DIABETES MELITUS PATIENTS at ABDUL WAHAB SJAHRANIE HOSPITALS, Proceeding on 5th Mulawarman Pharmaceutical Conferences, Mulawarman University, Samarinda, Indonesia, 23 – 24 April 2017.
- **Risna Agustina, Wisnu Cahyo Prabowo.** STUDY OF CHILDREN'S RELIABILITY PATTERN OF 0-5 YEARS AT ABDUL WAHAB SJAHRANIE HOSPITAL, Proceeding on 7th Mulawarman Pharmaceutical Conferences, Mulawarman University, Samarinda, Indonesia, 12 – 13 May 2018.

Patent Books:

Terminologi Medis

Research Interest:

- Complementary Medicine and Epidemiological Study

Research Grants:

- Research Grants for Lecturer of Faculty of Pharmacy in 2016 with title “STUDY OF INSULIN REGIMEN IN TYPE II DIABETES MELITUS PATIENTS at ABDUL WAHAB SJAHRANIE HOSPITALS”
- Research Grants for Lecturer of Faculty of Pharmacy in 2017 with title “STUDY OF CHILDREN'S RELIABILITY PATTERN OF 0-5 YEARS AT ABDUL WAHAB SJAHRANIE HOSPITAL”

Scholarship:

- Superior Scholarship for Master Program by Program Kalimantan Timur Cemerlang from year 2011-2013.
- Superior Scholarship for Profession Degree by Program Kalimantan Timur Cemerlang from year 2015-2016.

Biodata Partisipan

Nama	LIZDA ISWARI, S.T., M.Sc.
NIDN/NUP	0520077901
Perguruan Tinggi	Universitas Islam Indonesia
Program Studi	Teknik Informatika S-1
Jenis Kelamin	Perempuan
Jabatan Fungsional	ASISTEN AHLI

Pendidikan Tertinggi

S-2

Status Ikatan Kerja

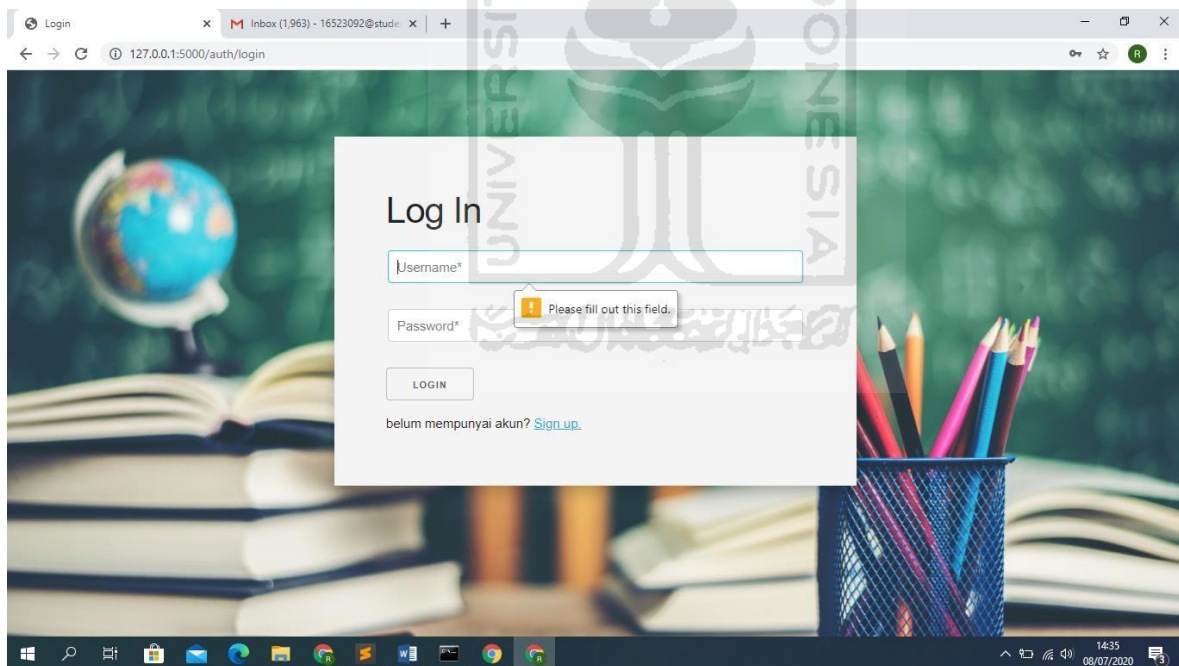
DOSEN TETAP

Status Aktivitas

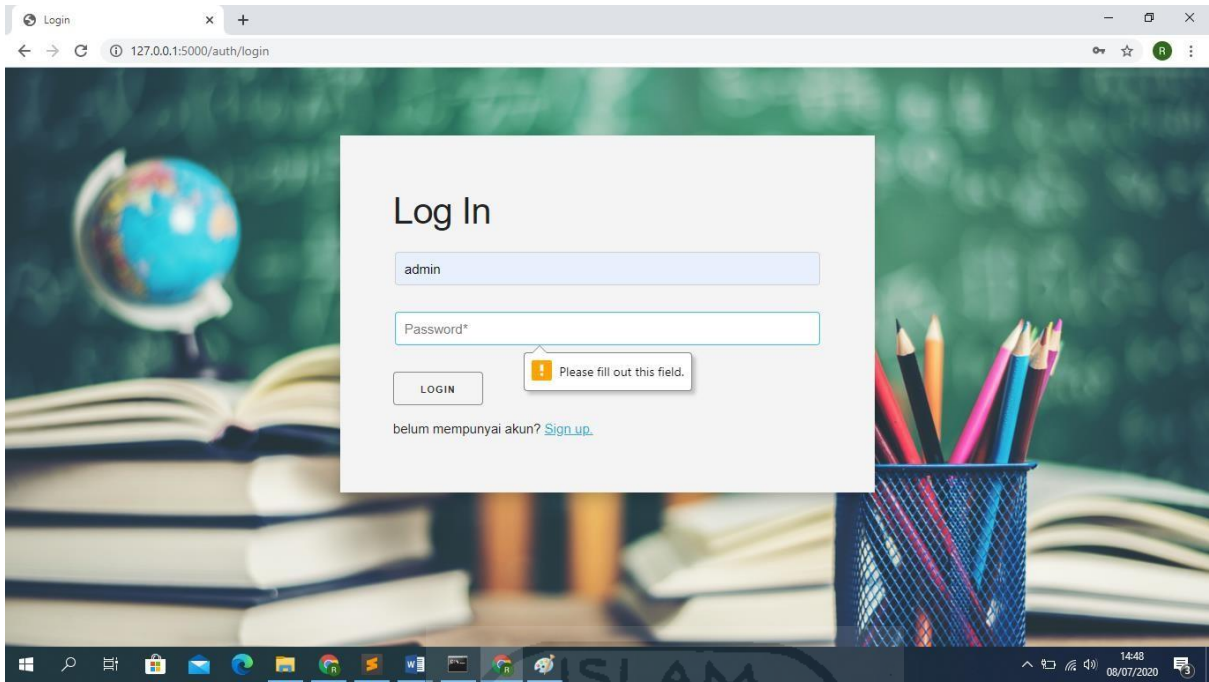
AKTIF MENGAJAR

Lampiran IV A Keterangan gambar hasil pengujian sistem

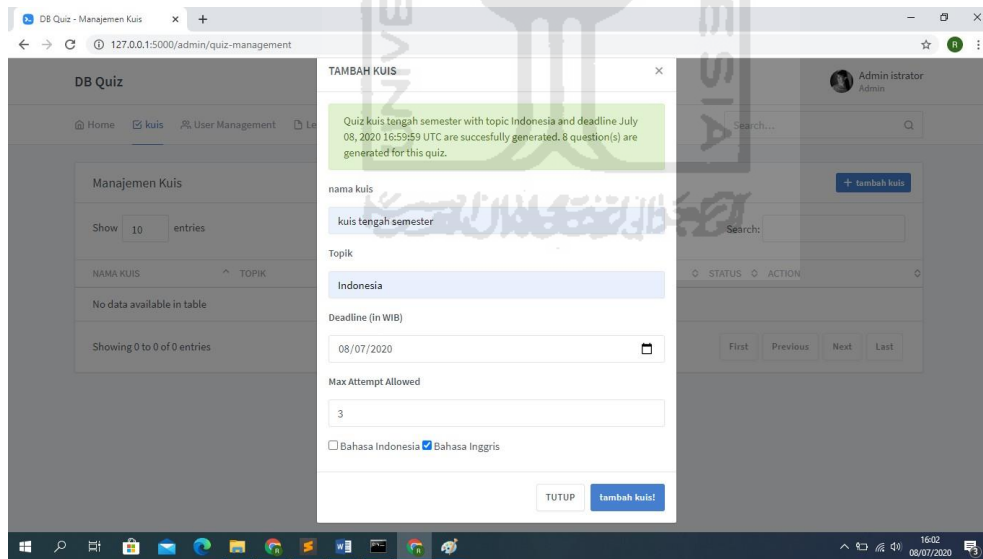
UC-1



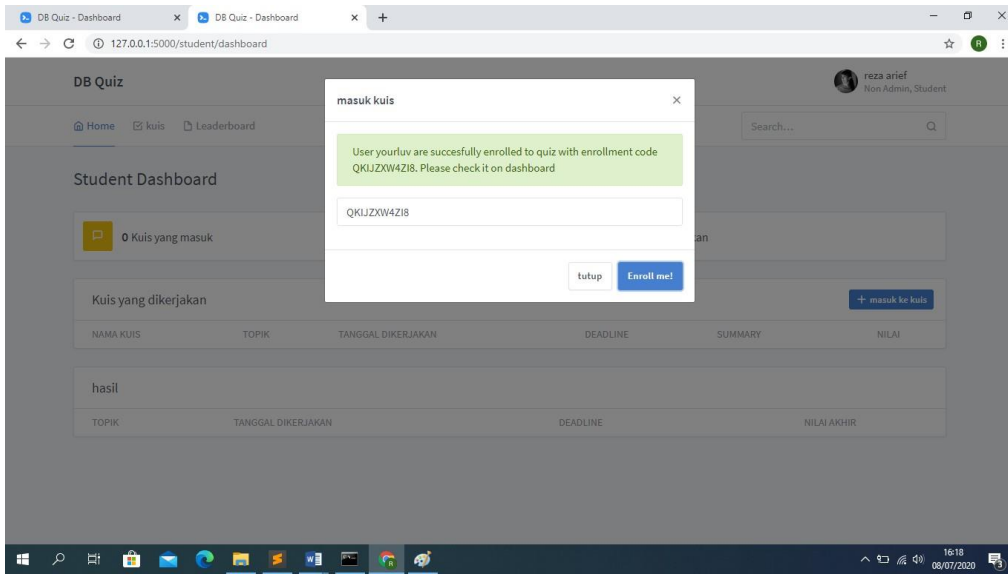
UC-2



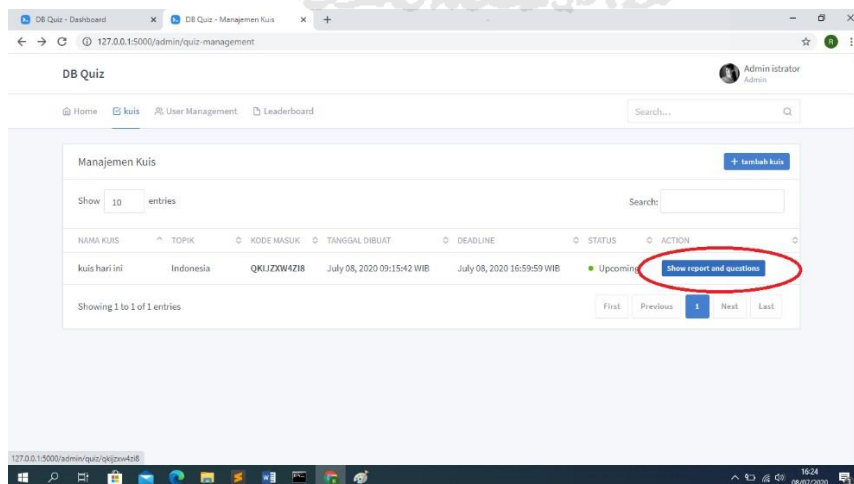
UC-3

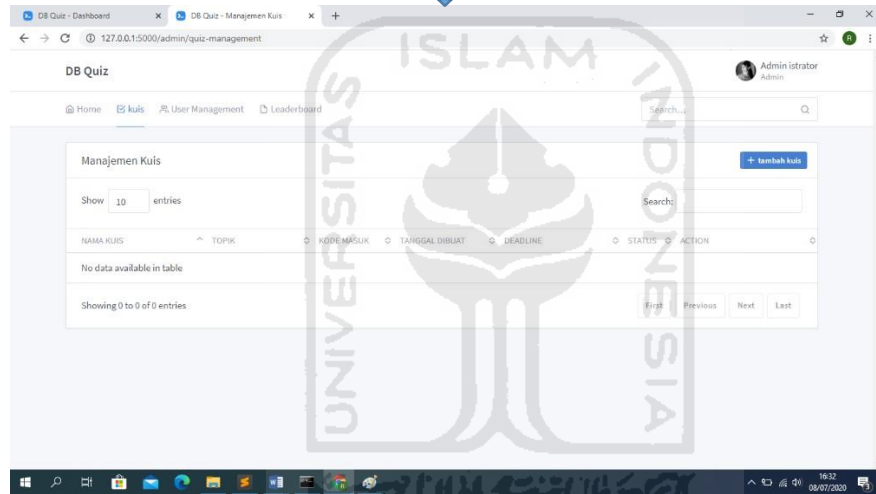
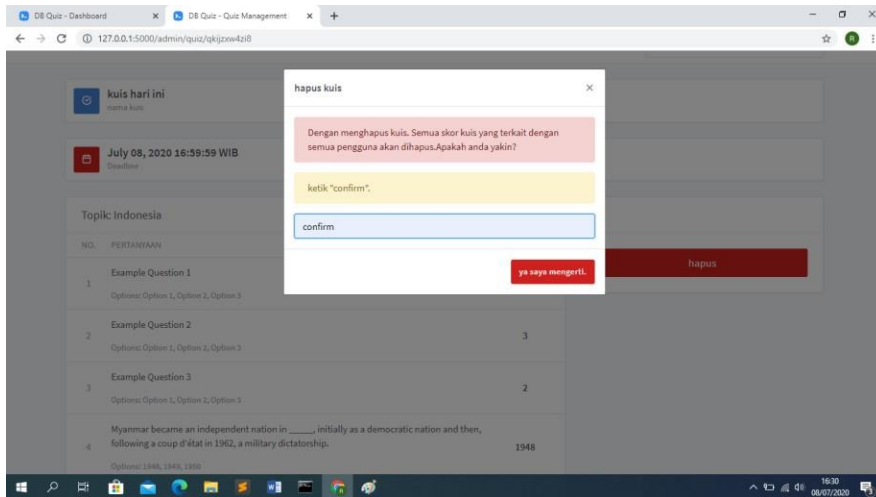


UC-4

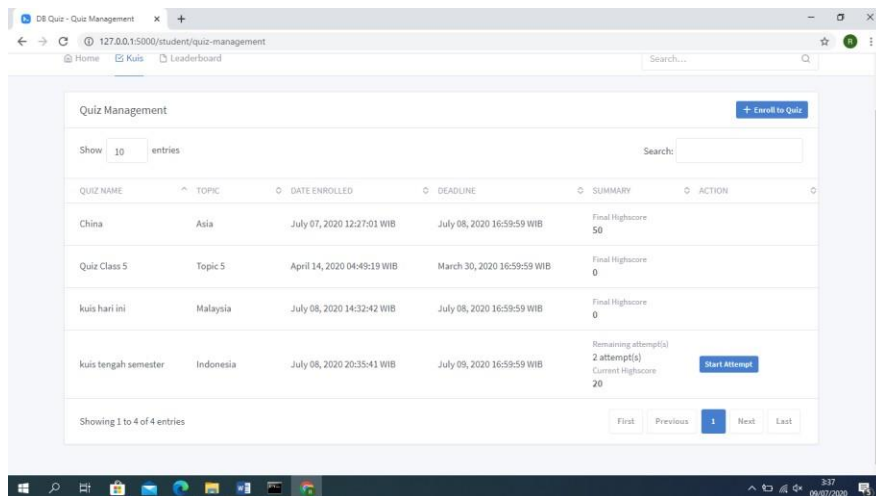


UC-5





UC-6



UC-7

Quiz Management + Enroll to Quiz

Show entries Search:

QUIZ NAME	TOPIC	DATE ENROLLED	DEADLINE	SUMMARY	ACTION
China	Asia	July 07, 2020 12:27:01 WIB	July 08, 2020 16:59:59 WIB	Final Highscore 50	
Quiz Class 5	Topic 5	April 14, 2020 04:49:19 WIB	March 30, 2020 16:59:59 WIB	Final Highscore 0	
kuis hari ini	Malaysia	July 08, 2020 14:32:42 WIB	July 08, 2020 16:59:59 WIB	Final Highscore 0	
kuis tengah semester	Indonesia	July 08, 2020 20:35:41 WIB	July 09, 2020 16:59:59 WIB	Remaining attempt(s) 0 attempt(s) Current Highscore 20	Continue

Showing 1 to 4 of 4 entries

First Previous **1** Next Last

UC-8

DB Quiz - Quiz Management x +

127.0.0.1:5000/admin/user-management

DB Quiz Admin istrator Admin

Home Kuis **User Management** Leaderboard Search...

User Management

Show entries Search:

USER ID	USERNAME	DISPLAY NAME (FIRST AND LAST NAME)	DATE CREATED	ACTION
5	desenfirman	Dese Firmansyah	April 14, 2020 04:51:50 WIB	Delete
2	member1	Membership	April 14, 2020 04:49:19 WIB	Delete
3	member2	Member2 ship2	April 14, 2020 04:49:19 WIB	Delete
4	member3	Member3 ship3	April 14, 2020 04:49:19 WIB	Delete

Showing 1 to 4 of 4 entries

First Previous **1** Next Last

UC-8

DB Quiz - Quiz Management x +

127.0.0.1:5000/admin/user-management

DB Quiz Admin istrator Admin

Home Kuis **User Management** Leaderboard Search...

User Management

Show entries Search:

USER ID	USERNAME	DISPLAY NAME (FIRST AND LAST NAME)	DATE CREATED	ACTION
5	desenfirman	Dese Firmansyah	April 14, 2020 04:51:50 WIB	Delete
2	member1	Membership	April 14, 2020 04:49:19 WIB	Delete
3	member2	Member2 ship2	April 14, 2020 04:49:19 WIB	Delete
4	member3	Member3 ship3	April 14, 2020 04:49:19 WIB	Delete

Showing 1 to 4 of 4 entries

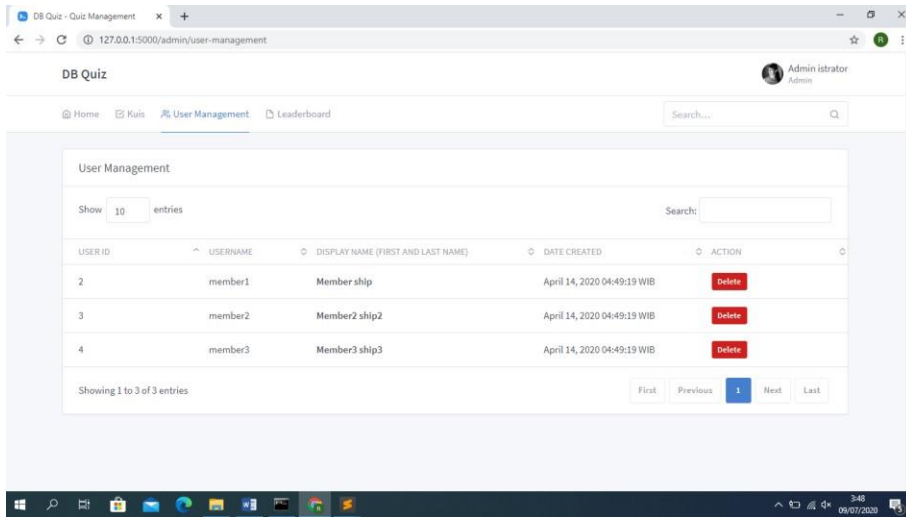
First Previous **1** Next Last

hapus User

By deleting user, All data and quiz report related to selected user will be deleted. This operation cannot be undone.

Masukkan nama username.

I understand all data related to this user will be deleted.



UC-9

