

**PENGEMBANGAN APLIKASI WEB *ONLINE SELF-
REGULATED LEARNING (ON-SRUII)*
DENGAN LARAVEL**



Disusun Oleh:

Nama : Andri Setyadi

NIM : 18523131

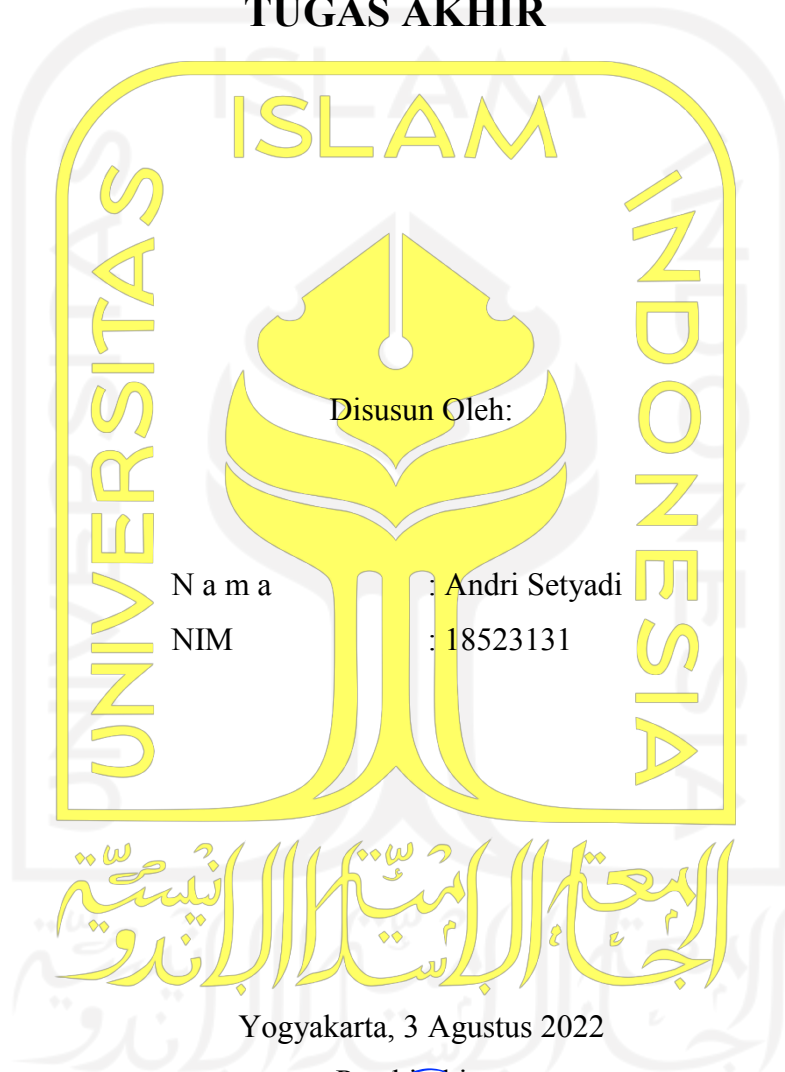
**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2022

HALAMAN PENGESAHAN DOSEN PEMBIMBING

**PENGEMBANGAN APLIKASI WEB *ONLINE SELF-
REGULATED LEARNING* (ON-SRUII)
DENGAN LARAVEL**

TUGAS AKHIR



Yogyakarta, 3 Agustus 2022

Pembimbing,

(Ahmad M. Raf'ie Pratama, S.T., M.I.T., Ph.D.)

HALAMAN PENGESAHAN DOSEN PENGUJI

**PENGEMBANGAN APLIKASI WEB *ONLINE SELF-REGULATED LEARNING* (ON-SRUII)
DENGAN LARAVEL**

TUGAS AKHIR

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

Yogyakarta, 3 Agustus 2022

Tim Penguji

Ahmad M. Rafie Pratama, S.T., M.I.T., Ph.D.

Anggota 1

Beni Suranto, S.T., M.SoftEng.

Anggota 2

Sri Mulyati, S.Kom., M.Kom.

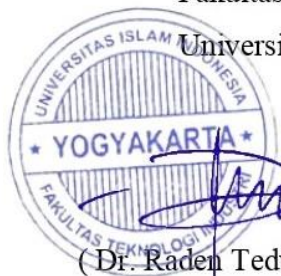


 Mengetahui

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 : Andri Setyadi

NIM : 18523131

Tugas akhir dengan judul:

**PENGEMBANGAN APLIKASI WEB *ONLINE SELF-
REGULATED LEARNING* (ON-SRUII)
DENGAN LARAVEL**

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila di kemudian 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 risiko dan konsekuensi apapun.

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

Yogyakarta, 3 Agustus 2022



(Andri Setyadi)

الجامعة الإسلامية

HALAMAN PERSEMBAHAN

Alhamdulillah, dengan mengucapkan puji syukur kepada Allah SWT atas rahmat, hidayah dan petunjuk-Nya sehingga saya dapat menyelesaikan tugas akhir ini.

Tugas akhir ini saya persembahkan untuk kedua orang tua dan kakak-kakak saya yang selalu mendukung, mendoakan dan menyemangati saya hingga sampai dititik ini. Dukungan dan semangat yang diberikan oleh keluarga adalah motivasi terkuat saya untuk menyelesaikan kuliah.

Saya juga mengucapkan terima kasih kepada dosen pembimbing yang selalu memberikan petunjuk dan arahan dalam pengerjaan tugas akhir ini.



HALAMAN MOTO

“Setiap masalah pasti ada jalan keluarnya”

--QS. Al Insyirah : 5-6--

“Kerjakan dan selesaikan apa yang sudah dimulai”

--Orang Tua--

“Semangat terus dan jangan mudah untuk menyerah”

--Orang Tua--



KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Segala puji syukur atas kehadiran Allah SWT yang telah melimpahkan rahmat, hidayah dan petunjuk-Nya sehingga saya dapat menyelesaikan tugas akhir yang berjudul “Pengembangan Aplikasi Web *Online Self-Regulated Learning* (ON-SRUII) Dengan Laravel”.

Dalam mengerjakan tugas akhir ini tidak terlepas dari bantuan banyak pihak sehingga pada kesempatan ini, penulis ingin mengucapkan terima kasih kepada semua pihak yang telah memberikan bantuannya dalam penyusunan tugas akhir sampai selesai. Terima kasih ini saya tujukan kepada:

1. Allah SWT yang telah memberikan rahmat, hidayah dan petunjuk-Nya sehingga saya dapat menyelesaikan tugas akhir ini.
2. Kedua orang tua saya, Bapak Suyadi dan Ibu Daryati dan kakak-kakak saya Teguh Sudarwanto dan Adityo Nugroho.
3. Prof. Fathul Wahid, S.T., M.Sc., Ph.D., Selaku Rektor Universitas Islam Indonesia.
4. Hari Purnomo, Prof., Dr., Ir., M.T., IPU, Selaku Dekan Fakultas Teknologi Industri, Universitas Islam Indonesia.
5. Hendrik, S.T., M.Eng., selaku Ketua Jurusan Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia.
6. Dr. Raden Teduh Dirgahayu, S.T., M.Sc., selaku Ketua Program Studi Jurusan Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia.
7. Ahmad M. Raf'ie Pratama, S.T., M.I.T., Ph.D, selaku Dosen Pembimbing Tugas Akhir dari Jurusan Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia.
8. Puji Rahayu, S.Pd., MLST, selaku Dosen yang melakukan penelitian bersama dari jurusan Pendidikan Bahasa Inggris, Fakultas Psikologi dan Ilmu Sosial Budaya, Universitas Islam Indonesia.
9. Bapak dan Ibu Dosen Jurusan Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia, yang telah mendidik dan memberikan ilmu selama masa perkuliahan.
10. Teman-teman seangkatan yang selalu belajar bersama.

Penulis berharap semoga laporan tugas akhir dan *website* yang telah dibuat dapat bermanfaat bagi dosen dan mahasiswa khususnya dari Program Studi Pendidikan Bahasa Inggris.

Yogyakarta, 3 Agustus 2022



(Andri Setyadi)



SARI

Pembelajaran pada mata kuliah *Research Instrument Development and Analysis (RIDA)* dilaksanakan secara daring akibat Covid-19. Pembelajaran daring dilakukan dengan dua cara, yaitu sinkron dan asinkron. Pada pembelajaran secara sinkron dosen akan langsung bertatap muka dengan para mahasiswa menggunakan aplikasi zoom, sedangkan pembelajaran secara asinkron dosen akan memberikan materi berupa video atau dokumen yang dikirim di classroom. Pada pembelajaran secara daring materi yang disampaikan dosen tidak dapat dibahas secara detail karena waktu yang terbatas.

Model pembelajaran secara *self-regulated learning* dirasa dapat menangani permasalahan-permasalahan yang dialami selama perkuliahan secara daring. *Self-regulated learning* adalah pembelajaran mandiri yang mahasiswanya akan mengatur, merencanakan dan mengelola sendiri kegiatan belajarnya untuk mendapatkan hasil yang maksimal.

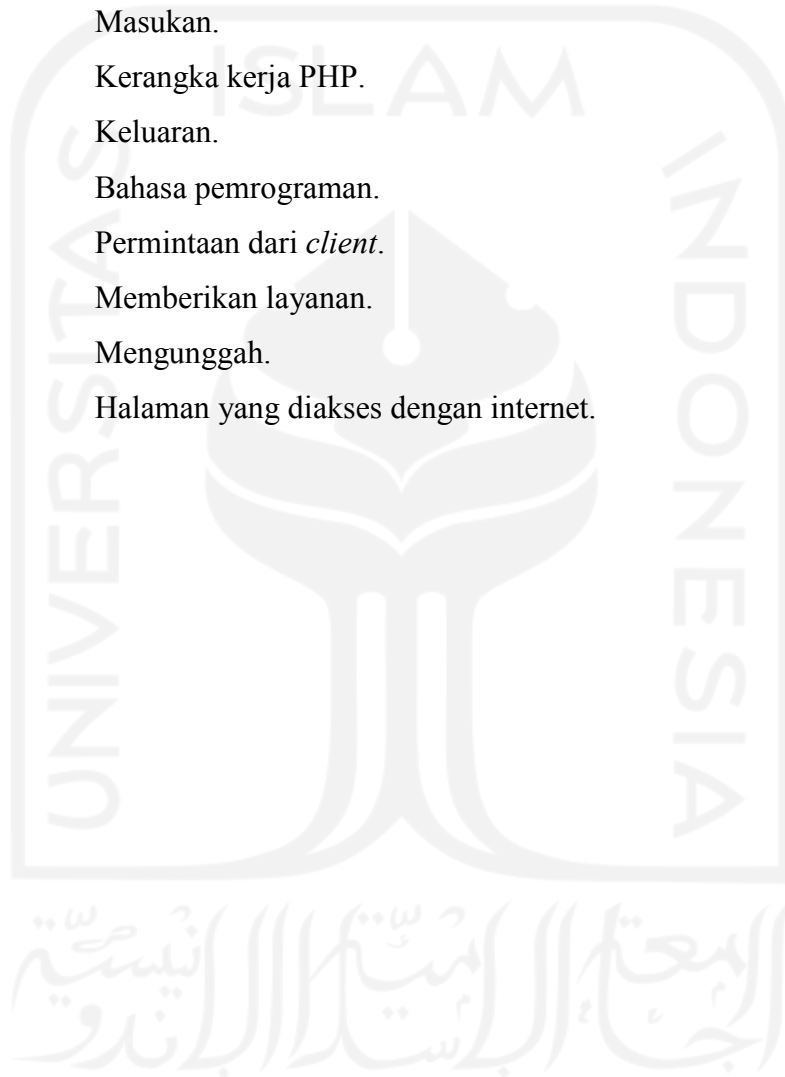
Pada penelitian ini peneliti menggunakan metode agile dengan *framework laravel* dalam pengembangan aplikasi web *online self-regulated learning (ON-SRUII)*. Pada metode pengembangan agile memiliki enam tahapan, yaitu *requirements, plan, design, develop, test,* dan *monitor*. Web akan diuji dari sisi fungsionalitas dengan cara dosen dan mahasiswa akan menjalankan web kemudian akan mengisi *form* pertanyaan.

Kesimpulan pada penelitian ini, berdasarkan pengujian kepada dosen dan mahasiswa aplikasi web ON-SRUII berhasil dibangun dengan dosen dapat memberikan materi dan melihat *progress* belajar dari mahasiswa dan mahasiswa dapat menyusun serta merencanakan sendiri kegiatan belajarnya.

Kata kunci: *self-regulated learning, agile, framework laravel, web*

GLOSARIUM

<i>Agile</i>	Metode pengembangan perangkat lunak.
<i>Backend</i>	Bagian <i>server</i> dan <i>database website</i> .
<i>Client</i>	Meminta layanan.
<i>E-Learning</i>	Sistem pembelajaran secara elektronik.
<i>Framework</i>	Kerangka kerja untuk pengembangan web.
<i>Input</i>	Masukan.
Laravel	Kerangka kerja PHP.
<i>Output</i>	Keluaran.
PHP	Bahasa pemrograman.
<i>Request</i>	Permintaan dari <i>client</i> .
<i>Server</i>	Memberikan layanan.
<i>Upload</i>	Mengunggah.
<i>Website</i>	Halaman yang diakses dengan internet.



DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING	ii
HALAMAN PENGESAHAN DOSEN PENGUJI	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR	iii
HALAMAN PERSEMBAHAN	v
HALAMAN MOTO	vi
KATA PENGANTAR	vii
SARI	ix
GLOSARIUM	x
DAFTAR ISI	xi
DAFTAR TABEL	xiii
DAFTAR GAMBAR	xiv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah	4
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian	4
1.6 Metodologi Penelitian <i>Backend</i>	5
1.7 Sistematika Penulisan	6
BAB II KAJIAN PUSTAKA	7
2.1 <i>E-Learning</i>	7
2.2 <i>Self-Regulated Learning</i>	8
2.3 <i>Backend Web Development</i>	9
2.3.1 Bahasa Pemrograman PHP	9
2.3.2 <i>Framework</i> Laravel	9
2.3.3 Basis Data MySQL	10
BAB III METODOLOGI PENELITIAN	11
3.1 Metodologi Penelitian	11
3.1.1 Metode Pengembangan	11
3.1.2 Prosedur Pengembangan	12
3.1.3 Langkah Scrum	14

3.2	Analisis Kebutuhan	18
3.2.1	Analisis Kebutuhan <i>Input</i>	19
3.2.2	Analisis Kebutuhan Proses	20
3.2.3	Analisis Kebutuhan <i>Output</i>	21
3.2.4	Analisis Kebutuhan <i>Tools</i>	22
3.2.5	Analisis Kebutuhan <i>Hosting</i>	22
3.3	Rancangan Sistem	23
3.3.1	Arsitektur Sistem	23
3.3.2	<i>Diagram</i> Alir Sistem	23
3.3.3	<i>Use Case Diagram</i>	25
3.3.4	<i>Activity Diagram</i>	26
3.3.5	<i>Diagram</i> Alir <i>Backend</i>	41
3.3.6	<i>Entity Relationship Diagram</i> (ERD)	42
3.3.7	Relasi Struktur <i>Database</i>	43
3.4	Pengerjaan Pengembangan <i>Website</i>	44
3.4	Desain Pengujian	45
BAB IV HASIL DAN PEMBAHASAN		49
4.1	<i>Develop Backend Website</i>	49
4.1.1	<i>Routes</i>	53
4.1.2	Dosen	54
4.1.3	Mahasiswa	56
4.2	<i>Hosting Website</i>	61
4.3	Tampilan <i>Website</i> ON-SRUII	63
4.4	Pengujian <i>Website</i> ON-SRUII	72
BAB V KESIMPULAN DAN SARAN		74
5.1	Kesimpulan	74
5.2	Saran	75
DAFTAR PUSTAKA		76

DAFTAR TABEL

Tabel 3.1 <i>Product Backlog</i>	14
Tabel 3.2 <i>Sprint 1</i> : Proses Penggunaan Akun UII	14
Tabel 3.3 <i>Sprint 2</i> : Proses Pengelolaan Materi Pembelajaran	15
Tabel 3.4 <i>Sprint 3</i> : Proses <i>Monitoring</i> Belajar Mahasiswa	15
Tabel 3.5 <i>Sprint 4</i> : Proses Mahasiswa Mengakses Materi	16
Tabel 3.6 <i>Sprint 5</i> : Proses Mahasiswa Belajar.....	16
Tabel 3.7 Desain pengujian <i>black box user</i> dosen	46
Tabel 3.8 Desain pengujian <i>back box user</i> mahasiswa	47
Tabel 4.1 Hasil pengujian fungsionalitas dari <i>user</i> dosen	72
Tabel 4.2 Hasil pengujian fungsionalitas dari <i>user</i> mahasiswa	73



DAFTAR GAMBAR

Gambar 1.1 Pengembangan ON-SRUII.....	3
Gambar 1.2 Metodologi penelitian <i>backend</i>	5
Gambar 2.1 Konsep MVC Laravel	10
Gambar 3.1 Metode agile.....	12
Gambar 3.2 Arsitektur <i>website</i>	23
Gambar 3.3 <i>Diagram</i> alir sistem.....	24
Gambar 3.4 <i>Use case diagram</i>	25
Gambar 3.5 <i>Activity diagram login</i> (Dosen).....	26
Gambar 3.6 <i>Activity diagram</i> tambah topik atau materi (Dosen)	27
Gambar 3.7 <i>Activity diagram update</i> topik atau materi (Dosen).....	28
Gambar 3.8 <i>Activity diagram</i> hapus topik atau materi (Dosen).....	29
Gambar 3.9 <i>Activity diagram</i> melihat <i>progress</i> mahasiswa (Dosen)	30
Gambar 3.10 <i>Activity diagram</i> membalas pesan (Dosen).....	31
Gambar 3.11 <i>Activity diagram login</i> (Mahasiswa)	32
Gambar 3.12 <i>Activity diagram</i> melihat topik (Mahasiswa).....	33
Gambar 3.13 <i>Activity diagram</i> melihat materi (Mahasiswa).....	34
Gambar 3.14 <i>Activity diagram</i> menambah <i>subgoal</i> (Mahasiswa).....	35
Gambar 3.15 <i>Activity diagram</i> menambah perencanaan (Mahasiswa).....	36
Gambar 3.16 <i>Activity diagram</i> menambah pemantauan (Mahasiswa)	37
Gambar 3.17 <i>Activity diagram</i> menambah strategi (Mahasiswa).....	38
Gambar 3.18 <i>Activity diagram</i> melihat <i>history</i> belajar (Mahasiswa)	39
Gambar 3.19 <i>Activity diagram</i> mengirim pesan (Mahasiswa).....	40
Gambar 3.20 <i>Diagram</i> alir <i>backend</i>	41
Gambar 3.21 <i>Entity Relationship Diagram</i> (ERD)	42
Gambar 3.22 Relasi struktur <i>database</i>	43
Gambar 4.1 <i>Install</i> laravel.....	49
Gambar 4.2 <i>Start project</i> laravel.....	49
Gambar 4.3 Tampilan XAMPP.....	50
Gambar 4.4 Tabel <i>database</i> onsruii	50
Gambar 4.5 Laravel <i>file .env</i>	50
Gambar 4.6 Laravel <i>folder controllers</i>	51
Gambar 4.7 Pengaturan <i>login</i>	51

Gambar 4.8 <i>Authorized redirect</i>	52
Gambar 4.9 <i>Client ID</i> dan <i>Client Secret</i>	52
Gambar 4.10 <i>services.php</i>	52
Gambar 4.11 Laravel <i>file .env</i>	52
Gambar 4.12 <i>Route</i> untuk <i>user</i> dosen	53
Gambar 4.13 <i>Route</i> untuk <i>user</i> mahasiswa.....	53
Gambar 4.14 Kode sumber untuk menambah topik	54
Gambar 4.15 Kode sumber untuk menambah materi	54
Gambar 4.16 Kode sumber untuk melihat <i>progress</i> mahasiswa.....	56
Gambar 4.17 Tampilan <i>progress</i> mahasiswa untuk dosen	56
Gambar 4.18 Kode sumber untuk <i>login</i> (belum pernah <i>login</i>)	57
Gambar 4.19 Kode sumber untuk <i>login</i> (sudah pernah <i>login</i>).....	57
Gambar 4.20 Halaman materi	58
Gambar 4.21 Kode sumber untuk menampilkan daftar materi.....	58
Gambar 4.22 Halaman belajar mahasiswa	58
Gambar 4.23 Kode sumber untuk mengambil materi	59
Gambar 4.24 Kode sumber untuk menampilkan <i>subgoals</i>	59
Gambar 4.25 Kode sumber untuk menampilkan materi terpilih.....	59
Gambar 4.26 Kode sumber untuk mengirim <i>chat</i>	59
Gambar 4.27 Halaman <i>history</i> belajar	59
Gambar 4.28 Kode sumber untuk menampilkan <i>history</i>	60
Gambar 4.29 Tampilan <i>form</i> SRL Palette	60
Gambar 4.30 Kode sumber untuk menyimpan isian SRL Palette	61
Gambar 4.31 <i>Database</i> di cPanel.....	61
Gambar 4.32 <i>File Manager</i> di cPanel.....	62
Gambar 4.33 Lokasi <i>folder public</i> laravel	62
Gambar 4.34 Lokasi <i>folder</i> onsruii	62
Gambar 4.35 Halaman beranda <i>website</i>	63
Gambar 4.36 Halaman <i>home</i> dosen	63
Gambar 4.37 Halaman <i>progress</i> mahasiswa.....	64
Gambar 4.38 Halaman tambah topik	64
Gambar 4.39 Halaman daftar topik.....	64
Gambar 4.40 Halaman <i>update</i> topik	65
Gambar 4.41 Halaman tambah materi	65

Gambar 4.42 Halaman daftar materi.....	65
Gambar 4.43 Halaman <i>update</i> materi	66
Gambar 4.44 Halaman pesan	66
Gambar 4.45 Halaman isian SRL Palette	66
Gambar 4.46 Halaman beranda <i>website</i>	67
Gambar 4.47 Halaman daftar topik.....	67
Gambar 4.48 Halaman daftar materi.....	67
Gambar 4.49 Halaman <i>history</i> belajar	68
Gambar 4.50 Tampilan daftar <i>subgoal</i>	68
Gambar 4.51 Tampilan <i>form subgoal</i>	68
Gambar 4.52 Tampilan <i>form Stating my understanding about the task</i>	69
Gambar 4.53 Tampilan <i>form What I have known about the topic</i>	69
Gambar 4.54 Tampilan <i>form assesing how well I understand this</i>	69
Gambar 4.55 Tampilan <i>form evaluating how well I already know about this content</i>	70
Gambar 4.56 Tampilan <i>form evaluating how well the content matches my current subgoals</i> 70	
Gambar 4.57 Tampilan <i>form take notes</i>	70
Gambar 4.58 Tampilan <i>form summarize</i>	71
Gambar 4.59 Tampilan <i>form make an inferences</i>	71
Gambar 4.60 Tampilan <i>form chat</i>	71

BAB

PENDAHULUAN

1.1 Latar Belakang

Belajar daring adalah kegiatan proses belajar yang memanfaatkan koneksi internet. Belajar daring di Universitas Islam Indonesia telah berlangsung hampir dua tahun dimulai pada akhir Maret 2020. Pembelajaran secara daring akibat dari pandemi *Covid-19* yang belum juga berakhir. Pada pembelajaran secara daring mahasiswa yang memegang peranan penting dalam mencapai keberhasilan belajar.

Model *self-regulated learning* adalah model pembelajaran yang siswanya sendiri akan mengatur secara efektif pembelajarannya dengan berbagai cara untuk memperoleh hasil belajar yang maksimal (Adhi et al., 2019). Belajar mandiri atau yang disebut dengan *self-regulated learning* adalah kegiatan belajar dalam memahami suatu materi yang dilakukan secara mandiri, pada kegiatan belajar mandiri mahasiswa dapat mengatur, merencanakan dan manajemen sendiri kegiatan belajarnya. Ada beberapa komponen dalam belajar secara mandiri, yaitu motivasi, metakognisi, kepercayaan asal pelajar, strategi belajar dan pengetahuan sebelumnya (Mukhid, 2008). Dengan *self-regulated learning* siswa akan mengaktifkan kognisi, perilaku, dan perasaan secara teratur dalam proses belajarnya untuk mencapai tujuan pembelajaran yang telah direncanakan (Aisyah, 2017).

Mata kuliah *Research Instrument Development and Analysis* (RIDA) adalah mata kuliah yang akan melatih mahasiswa untuk memahami jenis dan ciri-ciri penelitian kualitatif dan kuantitatif, cara menggunakannya dan cara mengembangkannya (berdasarkan Rancangan Pembelajaran Semester (RPS) mata kuliah). Mata kuliah yang berada di Program Studi Pendidikan Bahasa Inggris, Fakultas Psikologi dan Ilmu Sosial Budaya, Universitas Islam Indonesia.

Pelaksanaan pembelajaran mata kuliah RIDA saat ini dilakukan secara daring baik sinkron maupun asinkron. Untuk pembelajaran secara sinkron, mahasiswa akan langsung berinteraksi dengan dosen secara daring melalui aplikasi zoom, diskusi di Google doc atau WhatsApp, sedangkan pembelajaran secara asinkron mahasiswa akan diberikan materi dalam bentuk video atau pdf yang kemudian akan ada penugasan yang dibagikan di classroom.

Pemilihan mata kuliah RIDA yang akan dikembangkan ke aplikasi web, menurut informasi dari narasumber karena pada mata kuliah ini menitikberatkan pada praktek

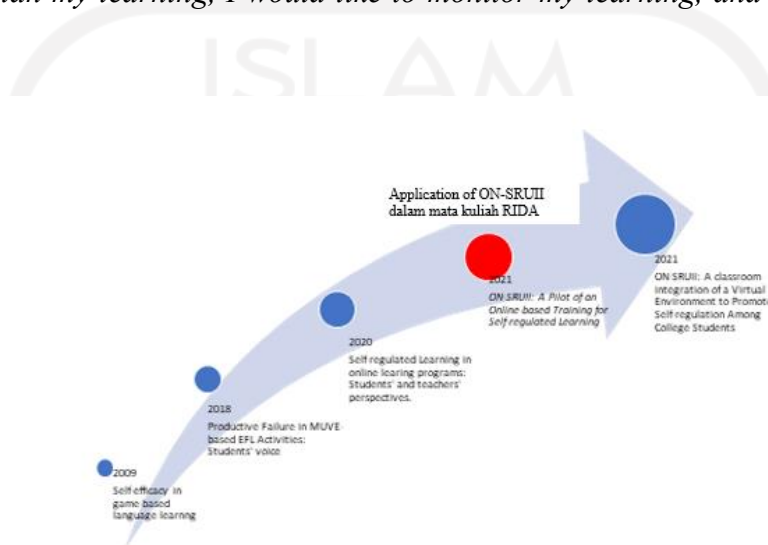
pengembangan *instrument* dan analisis kuantitatif. Menurut pengamatan narasumber mahasiswa dirasa kurang memiliki waktu yang cukup untuk berlatih desain dan menganalisis *instrument*. Narasumber menambahkan bahwa mahasiswa dan dosen pada matakuliah RIDA tidak memiliki waktu yang cukup untuk membahas secara detail dan mempraktekkan teori tentang dasar-dasar penyusunan *instrument* penelitian. Narasumber juga mengungkapkan bahwa keterampilan belajar mandiri dari para mahasiswa masih kurang, mengingat pembelajaran secara daring mengutamakan kemampuan belajar mandiri dari mahasiswa.

Pembelajaran secara *self-regulated learning* ini mahasiswa akan diminta untuk mengevaluasi sendiri terhadap proses belajarnya per materi. Hasil dari evaluasi yang dilakukan mahasiswa akan tersimpan di *database* yang kemudian dapat ditampilkan pada halaman dosen. Dosen juga memungkinkan untuk memberikan umpan balik terhadap isian dari para mahasiswa. Umpan balik tersebut bisa dalam bentuk komentar atau pesan yang kemudian oleh mahasiswa perlu diperhatikan.

Pengembangan web ini disesuaikan dengan CPMK pada mata kuliah RIDA, tepatnya pada CPMK kedelapan, yaitu keterampilan belajar mandiri. Sebagai upaya untuk membiasakan mahasiswa dapat belajar dengan memiliki tujuan pribadi dari masing-masing mahasiswa secara jelas dan selalu melakukan evaluasi terhadap keberhasilan dalam menggunakan strategi yang digunakan. Strategi pembelajaran kemandirian dengan menyediakan bahan ajar di Google Classroom belum mendapatkan hasil yang memadai karena mahasiswa masih harus sering diingatkan dan ditunjukkan mengenai bahan pembelajaran secara spesifik dengan tautan yang ditempelkan. Dengan strategi kemandirian ini diperlukan pemantauan dosen terhadap mahasiswa tentang semua aktivitas yang diberikan.

Pengembangan aplikasi ON-SRUII ini sebagai langkah untuk melatih keterampilan belajar dari para mahasiswa. Perancangan web ini diharapkan dapat mengatasi keterbatasan waktu belajar daring dan dapat melatih mahasiswa untuk membiasakan diri belajar sesuai dengan urutan, seperti pada konsep pengembangan belajar mandiri. Dimana urutan belajarnya dimulai dengan 1) Pemahaman terhadap tugas belajar, 2) Tujuan dan perencanaan yang telah ditentukan, 3) Taktik dari belajar dan 4) adaptasi. Dengan web ON-SRUII ini diharapkan dapat membuat mahasiswa untuk terbiasa merencanakan belajarnya secara jelas. Dengan ON-SRUII mahasiswa diberikan kesempatan untuk dapat berlatih lebih sering lagi dengan kecepatan yang dimiliki oleh masing-masing dari mahasiswa.

Sejarah pengembangan *Online Self Regulated* Universitas Islam Indonesia (ON-SRUII) saat ini sampai tahap yang keempat pada warna merah. Untuk gambarnya bisa dilihat pada Gambar 1.1, pada aplikasi-aplikasi yang telah dibuat terdahulu ada beberapa fitur yang dirasa dosen RIDA belum optimal salah satunya pada bagian SRL Palette. SRL Palette adalah bagian pada *website* yang akan digunakan untuk melatih pengetahuan mahasiswa secara mandiri terhadap materi yang sedang dipelajarinya. SRL Palette terdiri dari tiga bagian, yaitu *I would like to plan my learning, I would like to monitor my learning, and apply a learning strategy*.



Gambar 1.1 Pengembangan ON-SRUII

Pada penelitian yang sedang dijalankan ini akan mengembangkan suatu aplikasi web yang dapat digunakan oleh mahasiswa maupun dosen. Bagi mahasiswa dapat digunakan untuk mengembangkan keterampilan belajar mandiri dan melatih praktek penyusunan maupun analisis instrumen penelitian, baik kualitatif maupun kuantitatif. Pengembangan keterampilan dilakukan dengan cara mahasiswa akan menyusun tujuan pembelajarannya sendiri, kemudian akan mempelajari materi dan dilanjutkan dengan mengisi pertanyaan, mahasiswa juga dapat bertanya kepada dosen dengan fitur *chat*. Bagi dosen dapat melakukan *monitoring* atau pemantauan belajar mahasiswa, memberikan materi pembelajaran dan melihat semua isian yang dilakukan mahasiswa.

1.2 Rumusan Masalah

Rumusan masalah pada penelitian ini adalah berfokus pada pengembangan *backend* yang dapat memenuhi semua kebutuhan proses bisnis dan fungsionalitas dari web ON-SRUII.

1.3 Batasan Masalah

Penelitian ini merupakan bagian dari penelitian gabungan ON-SRUII. Pada penelitian ini berfokus pada pengembangan teknologi *backend* berbasis web dan *Database Management System* (DBMS).

1.4 Tujuan Penelitian

a. Tujuan Penelitian *Website* ON-SRUII

Untuk mengembangkan aplikasi web ON-SRUII yang dapat digunakan mahasiswa untuk belajar dari mana pun dan kapan pun, menyusun rencana belajar mahasiswa dan dapat mengatasi keterbatasan waktu belajar dengan dosen pada mata kuliah RIDA. Pada dosen dapat memberikan materi, *monitoring* dan mengevaluasi pembelajaran secara daring.

b. Tujuan Penelitian *Backend* ON-SRUII

Untuk mengembangkan *backend* yang mampu memenuhi semua kebutuhan proses bisnis dan fungsionalitas dari ON-SRUII.

1.5 Manfaat Penelitian

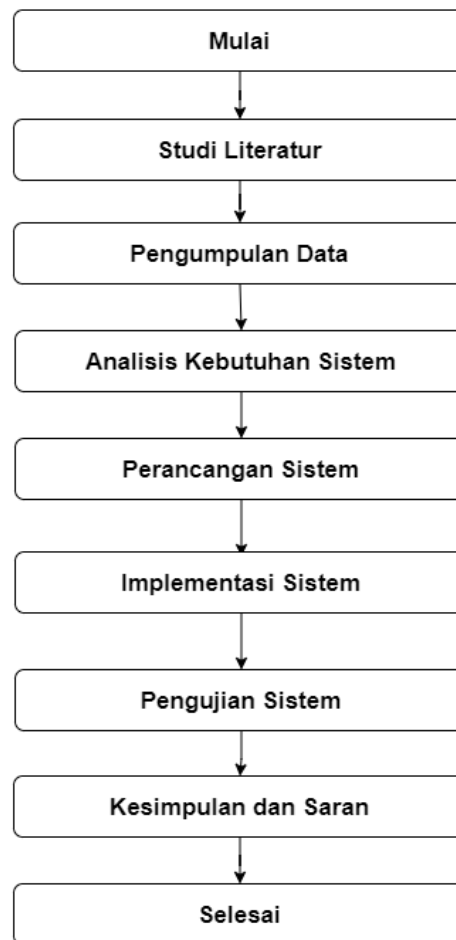
a. Manfaat Penelitian ON-SRUII

- Meningkatkan keterampilan belajar mandiri mahasiswa dengan membaca, mendengarkan, memahami dan menjawab pertanyaan.
- Dosen dapat memberikan materi dan memantau *progress* belajar dari para mahasiswa.
- Keterbatasan waktu saat belajar daring maupun luring dapat teratasi.
- Mahasiswa dapat belajar sesuai dengan materi yang ingin dipelajarinya sehingga menyesuaikan kondisi dari mahasiswa.

b. Manfaat penelitian *Backend* ON-SRUII

Manfaat dari penelitian ini adalah untuk memenuhi atau menyusun semua kebutuhan proses bisnis dan fungsionalitas dari *website*. Membuat Alur *website* dan penyimpanan data dapat berjalan dengan baik (sesuai dengan kebutuhan *user*) tanpa terjadinya *error* atau kesalahan.

1.6 Metodologi Penelitian *Backend*



Gambar 1.2 Metodologi penelitian *backend*

a. Studi literatur

Mengkaji berbagai jurnal dan artikel yang sesuai dengan topik penelitian untuk mendapatkan berbagai informasi mengenai permasalahan, metode dan solusi.

b. Pengumpulan data

Pengumpulan data dilakukan dengan wawancara kepada *client* untuk memperoleh berbagai data dan informasi yang terkait dengan sistem.

c. Analisis kebutuhan sistem

Melakukan analisis terhadap data yang diperoleh untuk mendapatkan kebutuhan dari sistem yang akan dibangun, agar sistem sesuai dengan yang dibutuhkan *client*.

d. Perancangan sistem

Melakukan perancangan sistem dari merancang *use case*, *activity diagram*, *database* dan merancang fungsi dari sistem.

e. Implementasi sistem

Pada tahap ini dilakukannya *develop* web, membuat *database* dan *backend* dari web di laravel.

f. Pengujian sistem

Pengujian sistem dilakukan untuk memastikan bahwa semua fungsionalitas sistem dapat berjalan sesuai dengan rancangan awal dan memastikan tidak terjadinya *error* pada sistem.

g. Kesimpulan dan saran

Kesimpulan berdasarkan dari penelitian yang telah dilakukan dan saran terhadap penelitian selanjutnya.

1.7 Sistematika Penulisan

BAB 1 PENDAHULUAN

Bab ini berisi latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian *backend* dan sistematika penulisan.

BAB II KAJIAN PUSTAKA

Bab ini berisi teori dan konsep dasar dari penelitian.

BAB III METODOLOGI PENELITIAN

Bab ini berisi langkah-langkah dalam mengembangkan aplikasi web yang terdiri dari metode, analisis, perancangan dan desain pengujian.

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi implementasi dan pengujian sistem berdasarkan rancangan yang telah dibuat.

BAB V KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dan saran tentang penelitian yang dilakukan.

BAB II KAJIAN PUSTAKA

2.1 *E-Learning*

E-Learning adalah metode pembelajaran yang memanfaatkan teknologi internet dalam kegiatan belajarnya. *E-Learning* merupakan suatu inovasi dari perkembangan teknologi internet untuk dunia pendidikan, yaitu pada proses pembelajaran. Pemanfaatan *E-Learning* bagi siswa dapat meningkatkan dan mengembangkan kemampuan belajar mandiri (Lutfiana & Wibawa, 2019).

E-Learning adalah sarana pendidikan yang menggabungkan motivasi diri, komunikasi, efisiensi dan teknologi menjadi satu kesatuan. Pemanfaatan *E-Learning* dapat meningkatkan efektifitas pembelajaran dan mengembangkan potensi yang ada pada diri siswa (Ma'ruf et al., 2017).

Penelitian yang terkait dengan *E-Learning* diantaranya:

- a. Permasalahan yang ditemukan peneliti dalam penelitiannya adalah keterbatasan waktu yang mengakibatkan guru tidak sempat memberikan latihan-latihan soal, permasalahan yang lain, yaitu ketika guru tidak dapat hadir untuk mengajar, maka materi akan disampaikan pada pertemuan selanjutnya, solusi yang diberikan peneliti, yaitu dengan membuat *E-Learning*. Metode pengembangan menggunakan agile dan pengujian menggunakan *black box testing* (Yauma et al., 2020).
- b. Pada penelitian ini peneliti mendapati permasalahan, yaitu proses kegiatan belajar antara guru dan siswa sangat terbatas. Peneliti memberikan solusi dengan merancang *E-Learning* berbasis web yang dapat diakses di mana pun dan kapan pun. Metode pengembangan waterfall dan pengujian *black box testing* (Susanto & Ayu, 2017).
- c. Pada penelitian ini peneliti menemukan permasalahan, yaitu proses kegiatan belajar yang belum maksimal karena masih menggunakan metode ceramah yang mengakibatkan siswa tidak tertarik dan kurang aktif dalam kegiatan pembelajaran. Solusi yang diberikan peneliti adalah dengan pengembangan media pembelajaran *E-Learning*. Metode pengembangan AADIE dan pengujian dengan penilaian angket (Lutfiana & Wibawa, 2019).

Pada ketiga penelitian tersebut lebih menyelesaikan permasalahan untuk mengatasi waktu kegiatan pembelajaran yang terbatas, kesulitan guru dalam memberikan materi ketika

tidak bisa hadir pada kegiatan pembelajaran karena ada halangan. Guru yang masih menggunakan metode pembelajaran tradisional, yaitu dengan ceramah. Solusi yang diberikan pada penelitian tersebut adalah mengembangkan media pembelajaran online atau *E-Learning*, tetapi aplikasi tersebut hanya berisikan pemberian materi, tugas dan nilai kepada siswa. Untuk pengumpulan data ketiganya sama menggunakan wawancara, observasi dan studi literatur.

2.2 *Self-Regulated Learning*

Self-Regulated Learning adalah pengelolaan pembelajaran yang dilakukan oleh siswa sendiri untuk meningkatkan pencapaian pembelajarannya. Pembelajaran secara mandiri ini dirasa perlu karena dapat mengembangkan kemandirian belajar siswa untuk menghadapi persaingan yang tinggi di dunia luar sehingga nantinya sudah terbiasa (Aminah & Maulida, 2020).

Model *Self-Regulated Learning* (SRL) adalah model pembelajaran yang siswanya sendiri akan mengatur secara efektif pembelajarannya dengan berbagai cara untuk memperoleh hasil belajar yang maksimal. Pemakaian model pembelajaran SRL dapat membuat siswa tidak mengalami kebosanan dan mengantuk saat pembelajaran, SRL juga dapat meningkatkan antusias belajar dari para siswa (Adhi et al., 2019).

Pembelajaran dengan menggunakan konsep *E-Learning* dapat mendukung berbagai cara belajar siswa yang berbeda-beda. Kemampuan pada komputer dan internet yang dapat memberikan pilihan atau keleluasaan kepada siswa untuk belajar sesuai dengan keinginannya atau rencannya dengan memanfaatkan fitur-fitur yang telah disediakan. Pembelajaran secara mandiri atau *self-regulated learning* dapat digunakan untuk menghadapi persaingan dan tantangan dimasa yang akan datang. Inilah mengapa pada level perguruan tinggi pembelajaran secara mandiri perlu ditingkatkan karena masih ditemukan berbagai persoalan yang terjadi, seperti penalaran pada suatu kejadian yang masih kurang dan mahasiswa yang tidak mampu untuk mengungkapkan gagasan atau menemukan masalah yang akan digunakan sebagai bahan tulisan (Ana & Achdiani, 2017).

Jadi, pada pembelajaran dengan model *self-regulated learning* dosen memberikan kebebasan kepada para mahasiswanya untuk dapat mengatur, merencanakan dan *monitoring* sendiri kegiatan belajarnya secara efektif untuk mendapatkan hasil yang optimal. Sesuai dengan kondisi masing-masing dari mahasiswa.

2.3 Backend Web Development

Backend Web Development adalah pengembangan yang berfokus pada sisi *server* serta mengelola dan membuat basis data. Membuat setiap *request* yang dilakukan *user* pada *browser* dapat berjalan dan dapat menampilkan data yang sesuai dengan seharusnya.

2.3.1 Bahasa Pemrograman PHP

PHP adalah bahasa pemrograman dibagian *server side* karena semua proses-proses yang terjadi tidak dijalankan pada sisi *client*, melainkan dijalankan pada sisi *server* (Hidayat et al., 2017). PHP pertama kali dikembangkan pada tahun 1994 oleh Rasmus Lerdorf.

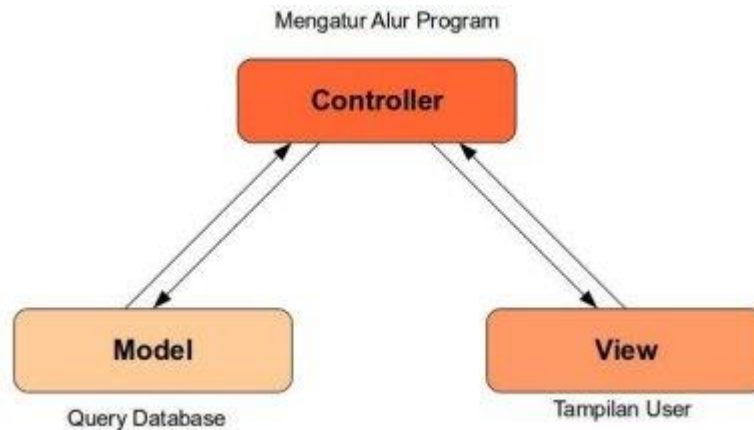
PHP digunakan dalam pembuatan aplikasi *website*, ini merupakan fungsi dari PHP sebagai bahasa pemrograman. PHP dapat berinteraksi dengan *folder file* dan *database*. PHP disebut sebagai bahasa *scripting*. PHP dapat dijalankan pada berbagai sistem operasi yang berbeda-beda, seperti (windows, mac dan linux) (Yuliano, 2017).

Keunggulan dari PHP adalah sifatnya yang *opensource* artinya pengguna dapat melakukan pengembangan dan modifikasi pada sistem atau aplikasi dengan bebas sesuai dengan kebutuhan dan keinginan dari para pengguna (Endra et al., 2021).

2.3.2 Framework Laravel

Laravel adalah kerangka kerja untuk membuat *website* berbasis PHP yang mempunyai sifat *opensource* dan gratis, dengan menggunakan konsep *model, view and controller* (MVC). Laravel dibuat oleh Taylor Otwell dan dirilis pada tahun 2011. *Model* adalah bagian yang menyimpan data untuk diambil oleh *controller*, kemudian akan dikirimkan ke *view*. *View* adalah bagian yang akan digunakan *user* untuk berinteraksi dengan *website*. *Controller* adalah bagian yang berfungsi sebagai penghubung antara *view* dengan *model*.

Waktu eksekusi pada laravel yang sedikit membuat proses loading *website* yang cepat sehingga laravel menjadi *framework* yang cocok digunakan pada pengembangan *website* (Somya & Nathanael, 2019).



Gambar 2.1 Konsep MVC laravel

Jadi, berdasarkan pada Gambar 2.1 ketika *user* melakukan aksi atau *request* pada *view* maka akan diteruskan ke *controller*, setelah sampai di *controller* ternyata *request* tersebut meminta data pada *database* sehingga akan diteruskan ke model untuk pengambilan datanya, setelah data diperoleh maka dikembalikan ke *controller* yang kemudian akan dikirim ke *view* untuk ditampilkan ke *user*.

2.3.3 Basis Data MySQL

MySQL adalah salah satu *database server* yang dalam pengaksesan basis datanya menggunakan bahasa *standard query language* (SQL) (Sofwan, 2011). Berikut adalah kelebihan menggunakan MySQL (Ma'ruf et al., 2017):

- a. Pada MySQL mendukung berbagai macam sistem operasi, seperti windows, Linux, FreeBSD Mac Os X server, dan Solaris.
- b. MySQL memiliki sifat *open source* atau gratis.
- c. MySQL bersifat *multi user* yang memungkinkan untuk digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa terjadinya masalah.
- d. Kecepatan dalam menangani *query* (perintah SQL) yang baik, yang memungkinkan SQL dapat diproses lebih banyak per satuan waktunya.

MySQL adalah salah satu perangkat lunak *Database Management System* (DBMS) atau sistem manajemen basis data yang menggunakan bahasa sql untuk mengubah dan mengelola pada *database*.

BAB III METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

3.1.1 Metode Pengembangan

Metode pengembangan aplikasi web pada penelitian ini menggunakan metode agile. Metode agile adalah metode pengembangan perangkat lunak yang mudah beradaptasi dengan perubahan karena memiliki berbagai prinsip pengembangan, perubahan tersebut terjadi dalam waktu yang relatif singkat (Rabbani & Krisnanik, 2020).

Prinsip-prinsip pada metode Agile sebagai berikut (Prihandani, 2016):

- a. *client* terlibat dalam pengerjaan sistem dari tahap awal sampai tahap akhir pengembangan.
- b. Komunikasi antar anggota tim yang terus dijaga agar pengembangan sistem dapat berjalan dengan baik.
- c. Anggota tim pengembangan akan berinteraksi antara satu dengan yang lainnya secara berkala agar pengerjaan sistem berhasil.
- d. *Client* akan diberikan informasi oleh pengembang mengenai tahapan pembangunan yang telah dikerjakan dalam periode waktu harian, mingguan atau bulanan.
- e. *Client* dan *user* tetap berkontribusi dalam pengembangan sistem.
- f. Perubahan kebutuhan dapat diselesaikan atau ditangani secara cepat.

Kelebihan dan kelemahan metode agile (Haryana, 2019):

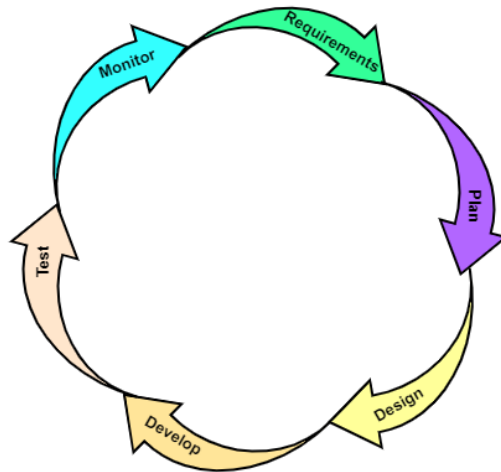
Kelebihan

- a. Kebutuhan dapat berubah sesuai dengan *client*.
- b. *Client* akan terlibat secara aktif.
- c. Perilisan sistem akan lebih cepat dan dilakukan secara berkala, disetiap akhir iterasi fungsi akan dirilis.
- d. Pengujian sistem akan dilakukan setiap saat.

Kekurangan

- a. Komunikasi yang dijalin dengan *client* kadang-kadang terlalu berlebihan.
- b. Pada proyek berskala besar metode ini sulit untuk diimplementasikan.
- c. Waktu yang singkat dalam merencanakan sistem.
- d. Memerlukan tim yang terlatih untuk manajemen pelaksanaan pengembangan.

Pada metode agile mempunyai enam proses, yaitu *Requirements*, *Plan*, *Design*, *Develop*, *Test*, and *Monitor*. Pemilihan untuk menggunakan metode agile karena *user* akan hampir terlibat di setiap tahapan proses pengembangan yang dilakukan, adanya fitur yang menjadi prioritas dalam pengembangan web. Metode dari agile digambarkan, seperti pada Gambar 3.1 (Vianna & Vianna, 2018).



Gambar 3.1 Metode Agile

3.1.2 Prosedur Pengembangan

Tahap-tahap proses pada agile sebagai berikut:

a. *Requirements*

Pada tahap ini mengumpulkan kebutuhan-kebutuhan dari *user* yang akan digunakan dalam pembuatan pada aplikasi web. Kebutuhan-kebutuhan ini berisikan proses bisnisnya, data-data yang akan dimasukkan, data yang akan ditampilkan dan proses yang akan dilakukan pada web.

b. *Plan*

Pada tahap ini akan dilakukan perencanaan mengenai pelaksanaan pengembangan sistem berdasarkan hasil dari *requirements*. Perencanaan tersebut diantaranya, yaitu menyusun rencana proses bisnis atau bagian yang akan dikerjakan terlebih dahulu, menentukan *tools* atau teknologi yang akan digunakan dalam pengembangan dan merencanakan cara pengerjaannya.

c. *Design*

Pada tahap ini akan membuat desain berdasarkan kebutuhan dari *user*. Ada beberapa desain yang akan dibuat, yaitu desain *database*, *use case* (menggambarkan proses-proses yang ada pada sebuah aplikasi) dan *activity diagram* (menggambarkan setiap proses yang ada pada *use case diagram*).

d. *Develop*

Pada tahap ini akan membuat kode pemrograman (*backend*) berupa fungsi-fungsi yang akan digunakan untuk menghubungkan *database* dengan tampilan web. Dapat menampilkan data-data yang tersimpan di *database* ke dalam *website*. Menggunakan *framework* laravel dengan bahasa PHP. Pada *database* menggunakan MySQL dengan XAMPP dan PHPMyadmin.

e. *Test*

Setelah web selesai maka akan dilakukan *test* atau pengujian. Tujuan dari pengujian ini adalah untuk memastikan semua fungsi dan fitur dapat berjalan sesuai dengan rancangan awal dan tidak ada *error*. Untuk pengujian dari *backend* sendiri akan menggunakan pengujian *black box testing*. Pengujian *black box* adalah pengujian yang dilakukan dari sisi fungsionalitas sistem.

f. *Monitor*

Langkah selanjutnya adalah *monitor*, pada langkah ini melakukan pemantauan terhadap aplikasi web. Melihat *feedback* dari pengguna yang memungkinkan bisa ditingkatkan atau memperbaiki *error* yang terjadi. Tujuan dari *monitor* adalah untuk menemukan celah-celah pada web yang memungkinkan untuk disempurnakan.

3.1.3 Langkah Scrum

1. Menentukan *Product Backlog*

Langkah pertama adalah menentukan prioritas dalam fitur. Daftar fitur-fitur yang akan dibangun berdasarkan skala prioritasnya terdapat pada tabel 3.1.

Tabel 3.1 *Product Backlog*

No	Deskripsi Fitur	Prioritas	Estimasi Waktu (hari)
1	Penggunaan akun UII	Sangat tinggi	5
2	Pengelolaan materi pembelajaran	Sangat tinggi	11
3	<i>Monitoring</i> belajar mahasiswa	Sangat tinggi	8
4	Mahasiswa mengakses materi	Sangat tinggi	4
5	Mahasiswa belajar	Sangat tinggi	14
Total			42

2. Menentukan *Sprint*

Pada langkah selanjutnya adalah melakukan *sprint* terhadap tabel 3.1. *Sprint* yang akan dilakukan berjumlah 5 dengan mempertimbangkan prioritas dan estimasi dari waktu pengerjaan yang telah dibuat.

Tabel 3.2 *Sprint 1* : Proses Penggunaan Akun UII

Fitur <i>Backlog</i>	<i>Task</i>	Estimasi Waktu (hari)
Penggunaan akun UII	<i>User</i> dapat <i>login</i> hanya dengan akun UII dan <i>user</i> dapat melakukan <i>logout</i> .	5
Total		5

Pada *sprint* pertama ini akan berfokus pada kebutuhan aplikasi untuk autentikasi dari *user*. *User* dosen maupun mahasiswa hanya bisa *login* dengan akun UII. Setelah berhasil *login* maka akan diarahkan ke halaman dosen untuk *user* dosen dan ke halaman mahasiswa untuk *user* mahasiswa.

Tabel 3.3 *Sprint 2* : Proses Pengelolaan Materi Pembelajaran

Fitur Backlog	Task	Estimasi Waktu (hari)
Pengelolaan materi pembelajaran	Dosen dapat menambahkan data topik.	3
	Dosen dapat memperbarui dan menghapus topik.	2
	Dosen dapat mengaktifkan dan menonaktifkan topik.	1
	Dosen dapat menambahkan data materi.	3
	Dosen dapat memperbarui dan menghapus materi.	2
	Total	11

Pada *sprint* kedua ini akan berfokus pada kebutuhan aplikasi untuk pengelolaan materi pembelajaran yang merupakan fitur dari halaman dosen. Untuk daftar bagian yang akan dikerjakan dapat dilihat pada tabel 3.3.

Tabel 3.4 *Sprint 3* : Proses *Monitoring* Belajar Mahasiswa

Fitur Backlog	Task	Estimasi Waktu (hari)
<i>Monitoring</i> belajar mahasiswa	Web dapat menampilkan mahasiswa yang telah belajar di setiap materinya.	3
	Web dapat menampilkan isian mahasiswa (<i>subgoals</i> , SRL Pallette dan pesan).	4
	Dosen dapat mengirim Pesan.	1
	Total	8

Pada *sprint* ketiga ini akan berfokus pada kebutuhan aplikasi untuk *monitoring* belajar dari mahasiswa. Fitur yang berada pada halaman dosen. Untuk daftar bagian yang akan dikerjakan terdapat pada tabel 3.4.

Tabel 3.5 *Sprint 4* : Proses Mahasiswa Mengakses Materi

Fitur Backlog	Task	Estimasi Waktu (hari)
Mahasiswa mengakses materi	Web dapat menampilkan topik.	2
	Web dapat menampilkan materi.	2
	Total	4

Pada *sprint* keempat ini akan berfokus pada kebutuhan aplikasi untuk mahasiswa dapat mengakses materi. Fitur yang berada di halaman mahasiswa. Untuk daftar bagian yang akan dikerjakan terdapat pada tabel 3.5

Tabel 3.6 *Sprint 5* : Proses Mahasiswa Belajar

Fitur Backlog	Task	Estimasi Waktu (hari)
Mahasiswa Belajar	Web dapat menampilkan data materi yang dipilih mahasiswa.	1
	Mahasiswa dapat menambahkan <i>subgoals</i> .	1
	Mahasiswa dapat mengkomplet <i>subgoals</i> .	1
	Mahasiswa dapat menambahkan <i>plan my learning</i> (ada 2 isian).	2
	Mahasiswa dapat menambahkan <i>monitor my learning</i> (ada 3 isian).	2
	Mahasiswa dapat menambahkan <i>apply a learning</i> (ada 4 isian).	2
	Mahasiswa dapat mengirim pesan.	3
	Web dapat menampilkan <i>history</i> belajar.	2
	Total	14

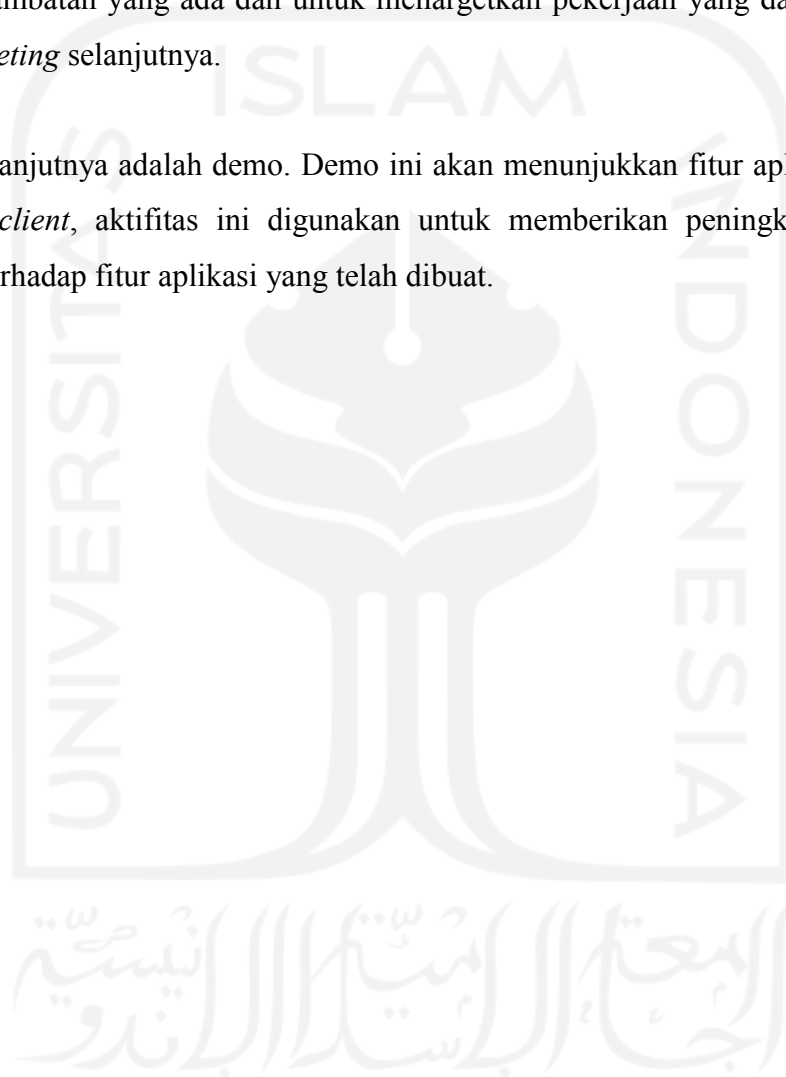
Pada *sprint* yang kelima ini akan berfokus pada kebutuhan aplikasi untuk dapat digunakan mahasiswa belajar. Untuk daftar bagian yang akan dikerjakan terdapat pada tabel 3.6.

3. Melakukan *Meeting*

Aktifitas selanjutnya adalah melakukan *meeting* atau pertemuan rutin yang dilakukan setiap minggu. Pertemuan ini untuk mengevaluasi mengenai bagian yang telah dikerjakan, masalah atau hambatan yang ada dan untuk menargetkan pekerjaan yang dapat diselesaikan untuk bahan *meeting* selanjutnya.

4. Demo

Aktifitas selanjutnya adalah demo. Demo ini akan menunjukkan fitur aplikasi yang telah dibuat kepada *client*, aktifitas ini digunakan untuk memberikan peningkatan dan untuk mengevaluasi terhadap fitur aplikasi yang telah dibuat.



3.2 Analisis Kebutuhan

Analisis kebutuhan adalah proses yang digunakan untuk mendapatkan kebutuhan dan spesifikasi sistem agar sesuai dengan keinginan atau yang dibutuhkan *user*. Analisis kebutuhan merupakan tahapan yang sangat penting dalam pengembangan sistem karena merupakan tahapan yang pertama untuk mengetahui bagaimana sistem ini akan dibangun. Pada penelitian ini menggunakan dua metode untuk mengumpulkan informasi tentang kebutuhan sistem, yaitu dengan wawancara dan studi literatur.

a. Wawancara

Wawancara adalah metode yang digunakan untuk memperoleh informasi dengan cara tanya jawab secara lisan. Wawancara dilakukan dengan dosen Pendidikan Bahasa Inggris mata kuliah RIDA dengan aplikasi zoom meeting. Kegiatan wawancara sendiri dilakukan lebih dari satu kali. Kegiatan wawancara yang dilakukan untuk mendapatkan informasi dan data-data yang terkait kebutuhan dari *user*. Hasil dari wawancara dengan dosen digunakan untuk membuat aplikasi pembelajaran berbasis web. Web tersebut diharapkan dapat menaikkan kemampuan belajar mandiri mahasiswa dari jurusan pendidikan Bahasa Inggris. Media pembelajaran berbasis web tersebut memiliki beberapa kriteria, kriterianya akan dijelaskan pada *subbab* analisis kebutuhan *input*, analisis kebutuhan proses, analisis kebutuhan *output*, dan analisis kebutuhan *hosting*.

b. Studi Literatur

Menganalisis berbagai jurnal artikel dan dokumen untuk mendapatkan informasi yang dapat digunakan pada penelitian ini. Informasi yang dimaksud adalah metode-metode yang dipakai, masalah yang diselesaikan dan solusi yang diberikan pada literatur tersebut.

Berikut adalah sumber data dan informasi pada penelitian yang dilakukan:

a. Dosen

Dosen berasal dari mata kuliah *Research Instrument Design and Analysis (RIDA)*, program studi Pendidikan Bahasa Inggris, Fakultas Psikologi dan Ilmu Sosial Budaya, Universitas Islam Indonesia.

b. Jurnal

Mencari jurnal-jurnal yang berkaitan dengan pengembangan aplikasi web *self-regulated learning*, metode pengembangan dan penyelesaian masalah.

3.2.1 Analisis Kebutuhan *Input*

Analisis kebutuhan *input* adalah analisis yang digunakan untuk memperoleh informasi mengenai kebutuhan *input* atau data yang akan dimasukkan oleh *user* ke sistem. Data *input* inilah yang nantinya akan disimpan ke *database*. Untuk mendapatkan informasi mengenai *input* yang dibutuhkan *user* maka dilakukan wawancara secara langsung dengan dosen menggunakan aplikasi zoom meeting dan ketika ada hal yang lupa ditanyakan saat zoom meeting maka akan bertanya lewat media *chatting*, yaitu whatsapp. Dengan metode wawancara secara langsung ini memudahkan untuk menggali informasi kepada *client* mengenai kebutuhan *input* yang dibutuhkan. Hasil dari wawancara dengan dosen didapatkan beberapa informasi diantaranya: ada dua *user* yang akan menggunakan web, yaitu dosen dan mahasiswa, *user* masuk harus dengan akun google UII, dosen akan memberikan materi pembelajaran, mahasiswa akan mengisi tujuan belajar, SRL Palette dan mahasiswa akan melakukan komunikasi di web. Untuk lebih lengkapnya di bawah ini merupakan hasil dari analisis untuk kebutuhan *input* yang akan ada pada web.

Berikut adalah hasil analisis untuk kebutuhan *input*:

- a. *Input* Dosen
 1. Data *login* akun Universitas Islam Indonesia
 - *Username*
 - *Password*
 2. Data topik
 - Nama topik
 - Tujuan
 3. Data materi
 - Judul
 - Isi materi
 - *Llink* youtube atau gambar
 4. Data pesan
 - Isi pesan
- b. *Input* Mahasiswa
 1. Data *login* akun Universitas Islam Indonesia
 - *Username*
 - *Password*
 2. Data *subgoals* (tujuan belajar).

3. *Data plan my learning*
 - *Stating my understanding about the task*
 - *What I have known about the topic*
4. *Data monitor my learning*
 - *Assesing how well I understand this*
 - *Evaluating how well I already know about this content*
 - *Evaluating how well the content matches my current subgoals*
5. *Data apply a learning strategy*
 - *Takes notes*
 - *Summarize*
 - *Make an inferences*
 - *Asking for help (untuk pesan)*

3.2.2 Analisis Kebutuhan Proses

Analisis kebutuhan proses adalah analisis yang digunakan untuk mengetahui berbagai tindakan atau pemrosesan yang akan digunakan untuk mengolah data pada sistem. Kebutuhan proses tersebut didapatkan berdasarkan hasil dari analisis kebutuhan *input* dan dari wawancara dengan dosen. Berdasarkan wawancara dengan dosen didapatkan informasi bahwa dari *user* dosen dapat memberikan materi pembelajaran lewat web dan dapat melihat riwayat belajar dari para mahasiswa. Dari sisi *user* mahasiswa dapat memilih materi dan mengisi pertanyaan yang ada di web dan dapat melihat riwayat belajarnya sendiri. Untuk lebih lengkapnya mengenai proses-proses yang akan ada pada sistem, di bawah ini merupakan hasil analisis untuk kebutuhan proses yang akan ada pada sistem.

Berikut adalah hasil dari analisis untuk kebutuhan proses pada web ON-SRUII:

- a. Dosen
 1. Proses *login* dan *logout*
 2. Proses menambahkan topik dan materi
 3. Proses menampilkan topik dan materi
 4. Proses menghapus topik dan materi
 5. Proses *update* topik dan materi
 6. Proses menampilkan *subgoals* mahasiswa
 7. Proses menampilkan *progress* mahasiswa (nama, NIM, waktu, *subgoals*)
 8. Proses menampilkan SRL Palette (*plan, monitor and strategy*) mahasiswa

9. Proses mengirim pesan
 10. Proses menampilkan pesan
- b. Mahasiswa
1. Proses *login* dan *logout*
 2. Proses menampilkan topik dan materi
 3. Proses menambahkan *subgoals*
 4. Proses menampilkan *subgoals*
 5. Proses untuk *complete subgoals*
 6. Proses menampilkan *plan my learning* (ada dua)
 7. Proses menambahkan *plan my learning* (ada dua)
 8. Proses *update plan my learning* (ada dua)
 9. Proses menampilkan *monitor my learning* (ada tiga)
 10. Proses menambahkan *monitor my learning* (ada tiga)
 11. Proses *update monitor my learning* (ada tiga)
 12. Proses menampilkan *apply a learning strategy* (ada empat)
 13. Proses menambahkan *apply a learning strategy* (ada empat)
 14. Proses *update apply a learning strategy* (ada empat)
 15. Proses mengirim pesan
 16. Proses menampilkan pesan
 17. Proses menampilkan *history* belajar (nama materi, waktu, *subgoals*, dan SRL Palette)

3.2.3 Analisis Kebutuhan *Output*

Analisis kebutuhan *output* adalah analisis yang digunakan untuk mendapatkan informasi atau luaran berdasarkan dari isian *user*. Kebutuhan *output* tersebut berdasarkan hasil dari analisis kebutuhan *input* dan dari wawancara dengan dosen. Bahwa dengan *input* yang dilakukan oleh *user* akan memberikan luaran, seperti di bawah ini.

Berikut adalah hasil dari analisis untuk kebutuhan luaran:

- a. Informasi mengenai *progress* belajar mahasiswa.
- b. Informasi mahasiswa yang telah belajar.
- c. Informasi terhadap isian SRL Palette.
- d. Informasi pesan dari mahasiswa terhadap materi yang belum paham.
- e. Informasi terhadap total waktu belajar setiap mahasiswa.
- f. Informasi mengenai rencana *subgoal* yang dibuat mahasiswa.

3.2.4 Analisis Kebutuhan *Tools*

Analisis kebutuhan *tools* adalah analisis yang digunakan untuk mendapatkan informasi mengenai perangkat lunak yang dapat dimanfaatkan dalam pengembangan sistem. Untuk mendapatkan informasi tersebut menggunakan studi literatur. Dalam pencarian yang dilakukan didapatkan informasi sebagai berikut: membutuhkan perangkat lunak yang dapat digunakan sebagai editor penulisan kode php, perangkat lunak yang dapat digunakan untuk mendesain diagram, *framework* yang dapat digunakan sebagai *backend*, dan perangkat lunak yang dapat menjadi *server* di *local*. Untuk lebih jelasnya di bawah ini adalah hasil dari analisis kebutuhan *tools*.

Berikut adalah hasil analisis untuk kebutuhan *tools* yang akan digunakan:

- a. Sublime
- b. Xampp
- c. *Framework* laravel
- d. Draw.io
- e. Chrome

3.2.5 Analisis Kebutuhan *Hosting*

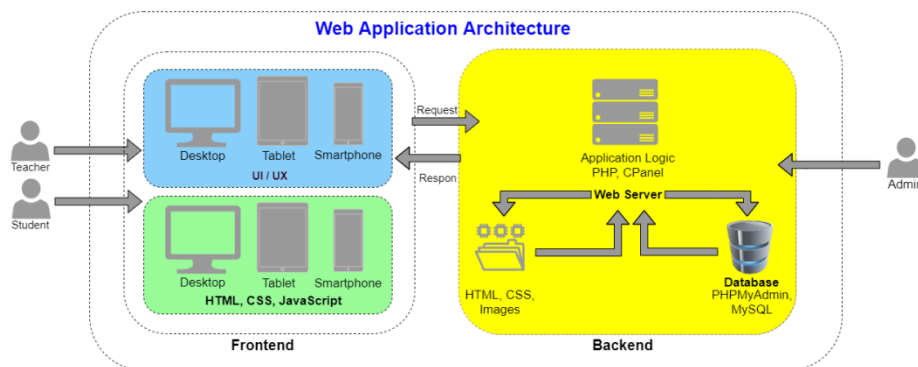
Analisis kebutuhan *hosting* adalah analisis yang digunakan untuk mendapatkan informasi mengenai layanan internet yang akan digunakan untuk mengelola web, yang sesuai dengan sistem. Penggunaan *hosting* ini agar web yang dibuat dapat diakses oleh *user* dari mana pun dan kapan pun. Kebutuhan *hosting* didapatkan dari wawancara dengan dosen dan berdasarkan spesifikasi dari sistem. Berdasarkan wawancara dengan dosen didapatkan bahwa untuk sementara 6 bulan, dapat diperpanjang dan dapat ditingkatkan layanannya. Lebih lengkapnya di bawah ini adalah hasil analisis untuk kebutuhan *hosting* yang akan digunakan pada web.

Berikut adalah hasil analisis untuk kebutuhan *hosting* yang akan digunakan:

- a. Pembuatan web menggunakan *framework* laravel.
- b. *Database* menggunakan MySQL.
- c. Diakses oleh 100-150 siswa.
- d. Menyimpan materi dalam jumlah banyak.
- e. Memakai nama domain onsruii.com.
- f. Menyediakan CPanel.

3.3 Rancangan Sistem

3.3.1 Arsitektur Sistem



Gambar 3.2 Arsitektur *website*

Gambar 3.2 merupakan diagram arsitektur *website*. Mempunyai tiga bagian, yaitu UI/UX, *Frontend* dan *Backend*. Pada penelitian ini peneliti mengerjakan pada bagian *Backend*. Pada bagian *backend* mengerjakan fungsi-fungsi untuk menangani setiap perintah dari *user*, merancang dan membuat *database*, membuat kode untuk mengambil data pada *database* yang akan ditampilkan ke *website* dan melakukan *hosting* pada *website* agar bisa diakses dari mana pun oleh mahasiswa maupun dosen dengan menggunakan koneksi internet.

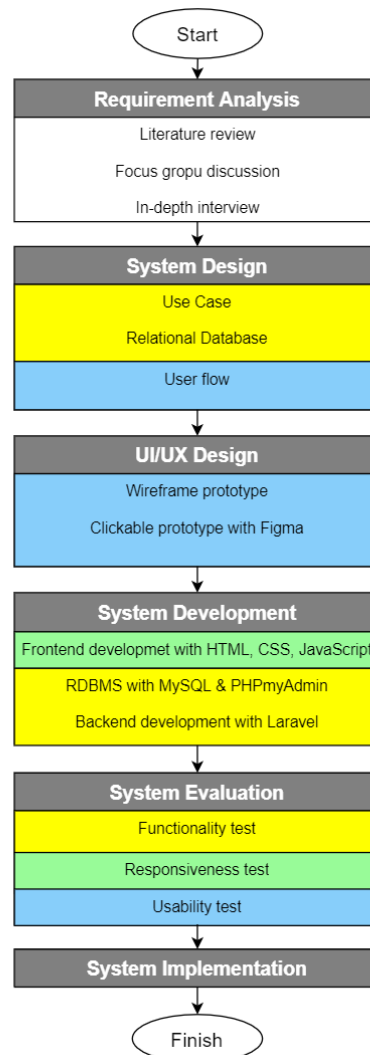
3.3.2 Diagram Alir Sistem

Pada tahap pengembangan sistem ini dimulai dengan menganalisis kebutuhan, analisis kebutuhan terdiri dari tinjauan pustaka, diskusi dan wawancara. Setelah analisis kebutuhan dilanjutkan dengan mendesain sistem, yang terdiri dari mendesain *use case*, *database* dan *user flow*. Setelah mendesain sistem selanjutnya mendesain tampilan di tahap desain UI/UX yang terdiri dari membuat *wireframe prototype* dan *prototype* yang bisa dijalankan. Hasil dari desain sistem dan desain UI/UX akan dibangun pada tahap *system development* yang terdiri dari membuat *database* menggunakan RDBMS MySQL di phpMyAdmin, membuat *backend* dengan *framework* laravel dan membuat *frontend* dengan HTML, CSS dan JavaScript. Setelah di *develop* maka dilakukan *system evaluation*, yang terdiri dari *functionality test*, *responsiveness test*, and *usability test*. Langkah tersebut digunakan untuk menguji web agar sesuai dengan kebutuhan *user* dan tidak ada kesalahan. Langkah terakhir adalah *system implementation*.

Berikut adalah gambar diagram alir sistem :

Keterangan Warna pada Diagram Alir

UI/UX
 Frontend
 Backend
 Bersama

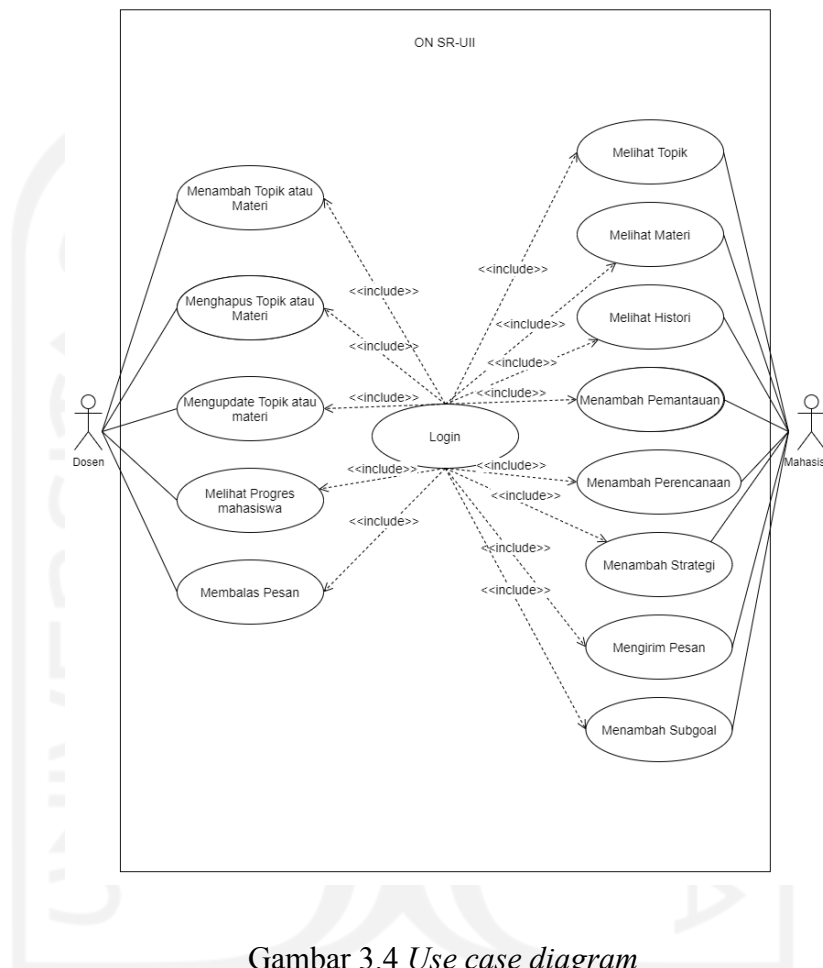


Gambar 3.3 Diagram alir sistem

Pada penelitian ini, yang merupakan bagian dari *backend* adalah warna kuning yang diantaranya, pada bagian *system desain* ada *use case* dan *relational database*, bagian *system development* ada *RDBMS with MySQL & PHPmyAdmin* dan *backend development with laravel*, dan pada bagian *system evaluation* ada *functionality test*. Pada bagian *requirement analysis* semua anggota tim melakukan *requirement analysis*, untuk mendapatkan kebutuhan atau informasi yang akan digunakan untuk pengembangan *website* berdasarkan bagian (UI/UX, *frontend*, dan *backend*) yang telah ditentukan.

3.3.3 Use Case Diagram

Berikut ini merupakan *use case diagram* dari aplikasi web *online self-regulated learning*. *Use case diagram* adalah bagan untuk mengetahui fungsional sistem dan untuk menjelaskan interaksi antara aktor dengan sistem (Yauma et al., 2020).



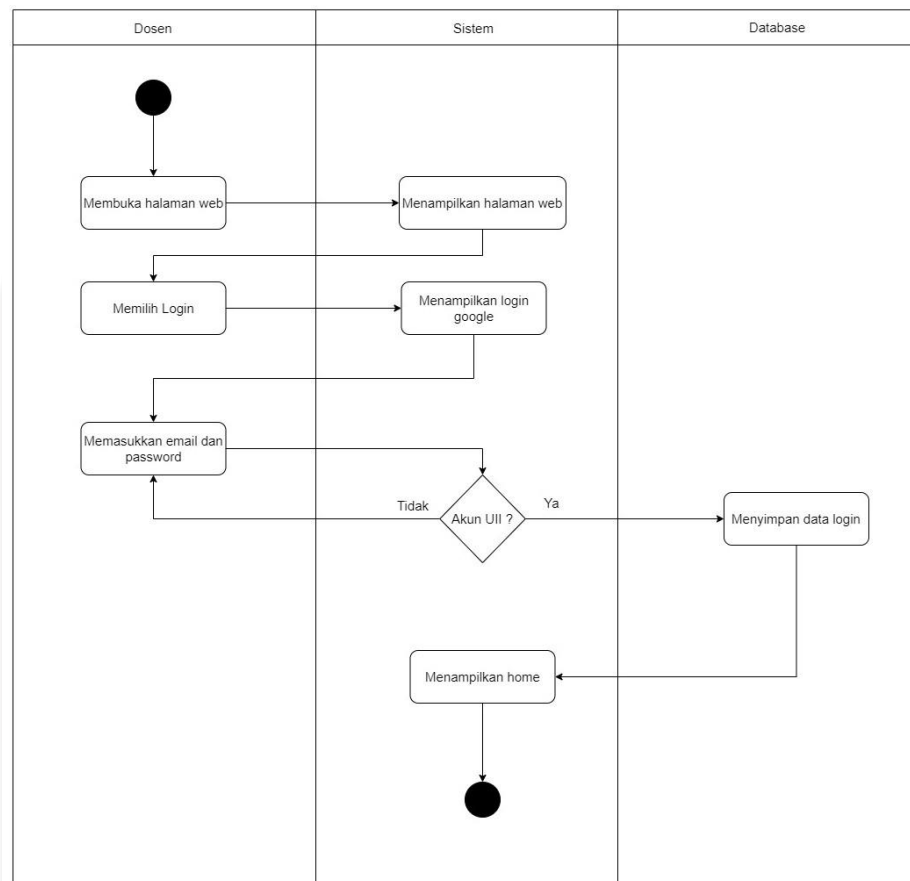
Gambar 3.4 Use case diagram

Pada *use case* terdapat dua aktor, yaitu aktor dosen dan aktor mahasiswa. Aktivitas yang bisa dilakukan oleh kedua aktor berbeda dan ada satu aktivitas yang sama, yaitu pada *login*. Pada aktor dosen dapat melakukan 6 aktivitas, yaitu *login*, menambah topik atau materi, menghapus topik atau materi, *update* topik dan materi, melihat *progress* mahasiswa, dan membalas pesan, sedangkan pada aktor mahasiswa dapat melakukan 9 aktivitas, yaitu *login*, melihat topik, melihat materi, melihat *history*, menambah pemantauan, menambah perencanaan, menambah strategi, mengirim pesan dan menambah *subgoal*.

3.3.4 Activity Diagram

a. Activity Diagram Dosen

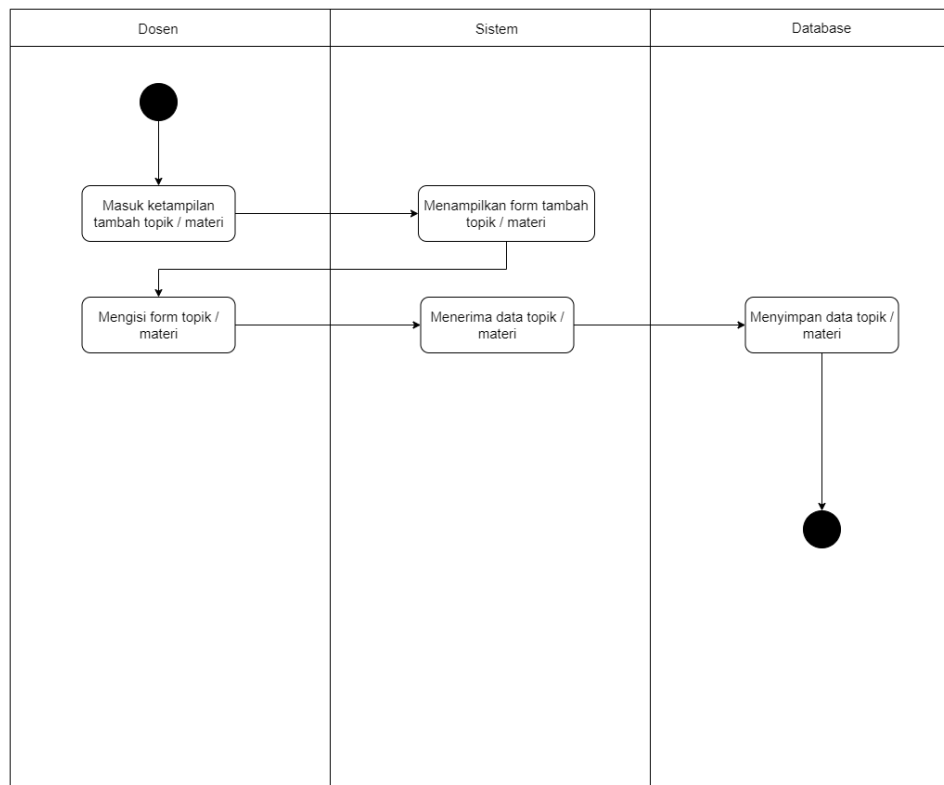
1. Login



Gambar 3.5 Activity diagram login

Ini merupakan *activity diagram login*. Activity ini harus dilakukan oleh dosen untuk dapat masuk ke halaman dosen. Pertama membuka alamat web kemudian memilih *login*, setelah memilih *login* masukkan alamat *email* UII kemudian akan diarahkan ke *login gateway* UII kemudian mengisi *username* dan *password* akun UII. Jika yang dimasukkan bukan akun UII maka tidak bisa masuk ke halaman dosen pada *website* dan akan mengulangi untuk mengisi *email* lagi.

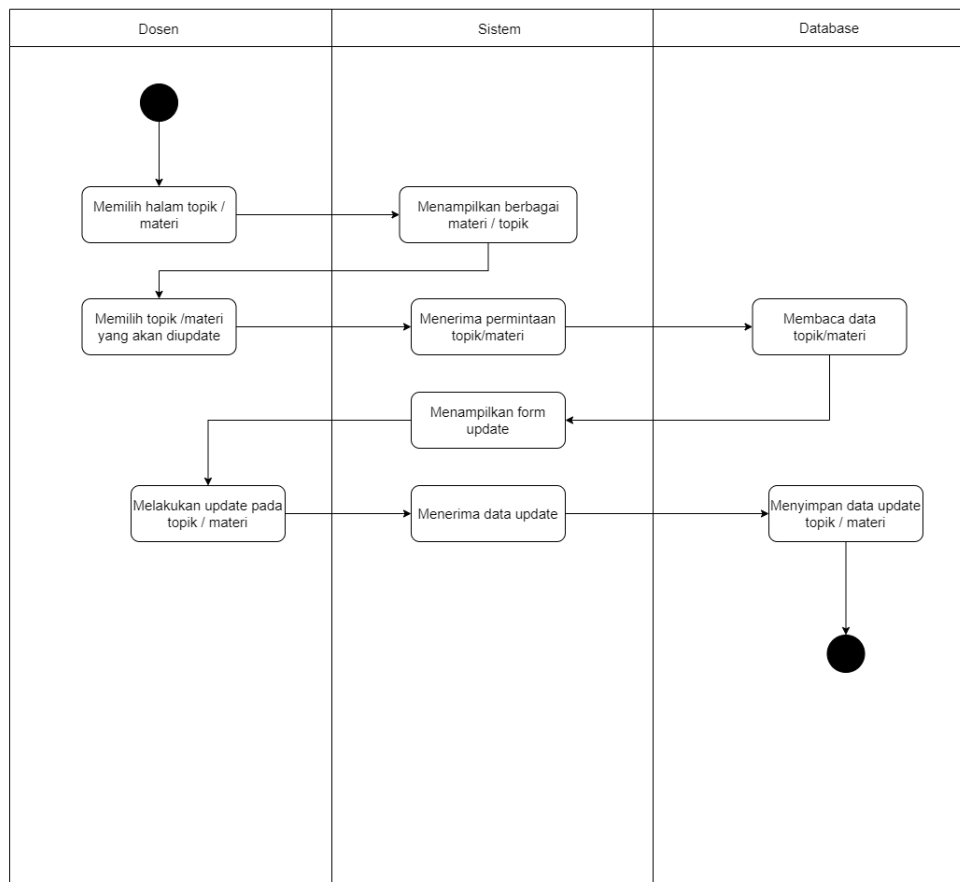
2. Tambah Topik atau Materi



Gambar 3.6 *Activity diagram* tambah topik atau materi

Ini merupakan *activity diagram* untuk menambah topik atau materi. Pada *activity* ini akan dilakukannya penambahan topik baru atau penambahan materi baru. Penambahan dimulai dengan masuk ke halaman tambah topik atau materi, kemudian sistem akan menampilkan *form* untuk topik atau materi, kemudian dosen akan melakukan pengisian pada *form* tersebut. Pada *form* tambah topik, data yang ditambahkan, yaitu nama topik dan tujuan belajar, sedangkan pada *form* tambah materi data yang ditambahkan, yaitu nama materi, isi materi, *link* video atau gambar. Jika semua sudah terisi maka dosen dapat klik tambah pada *website* yang kemudian akan diproses oleh sistem untuk disimpan ke dalam *database* topik atau *database* materi.

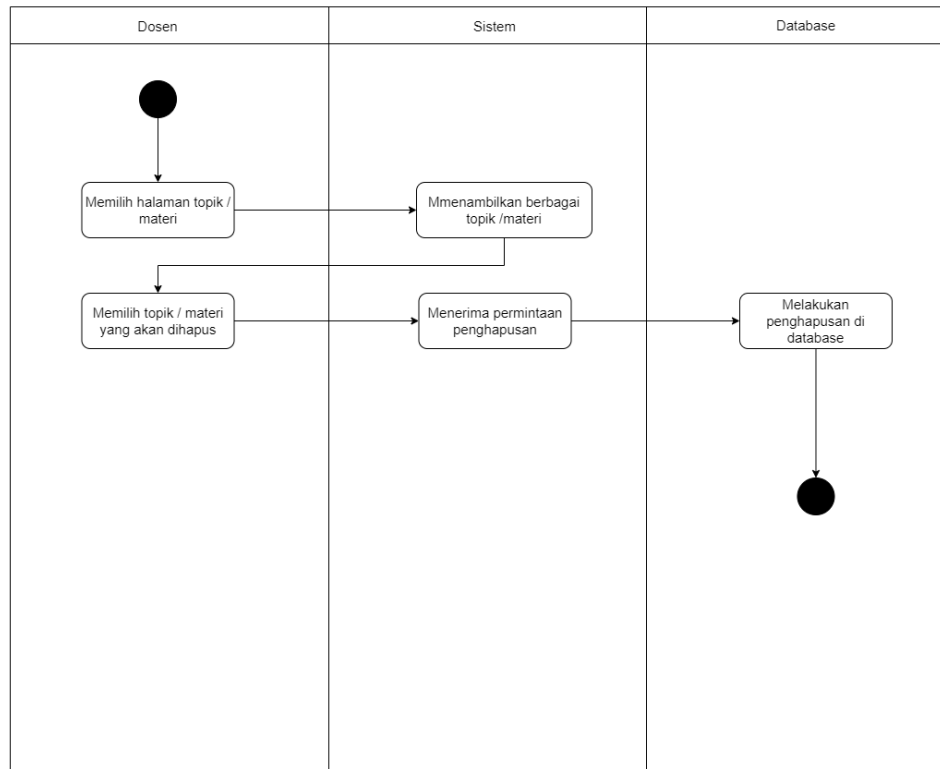
3. Update Topik atau Materi



Gambar 3.7 Activity diagram update topik atau materi

Ini merupakan *activity diagram update* topik atau materi. Pada *activity* ini dosen dapat melakukan perubahan terhadap data topik atau materi yang telah dibuat sebelumnya. *Activity* ini dimulai dengan dosen masuk ke halaman topik atau materi, kemudian sistem akan menampilkan daftar topik atau materi yang ada, kemudian dosen akan memilih topik atau materi yang akan diubah datanya dengan menekan *button update* pada topik atau materi tersebut, setelah topik atau materi dipilih sistem akan menampilkan data mengenai topik atau materi yang dipilih ke *form update*, kemudian dosen dapat melakukan perubahan pada data tersebut. Setelah dosen selesai melakukan perubahan maka sistem akan menyimpan perubahan tersebut ke *database*.

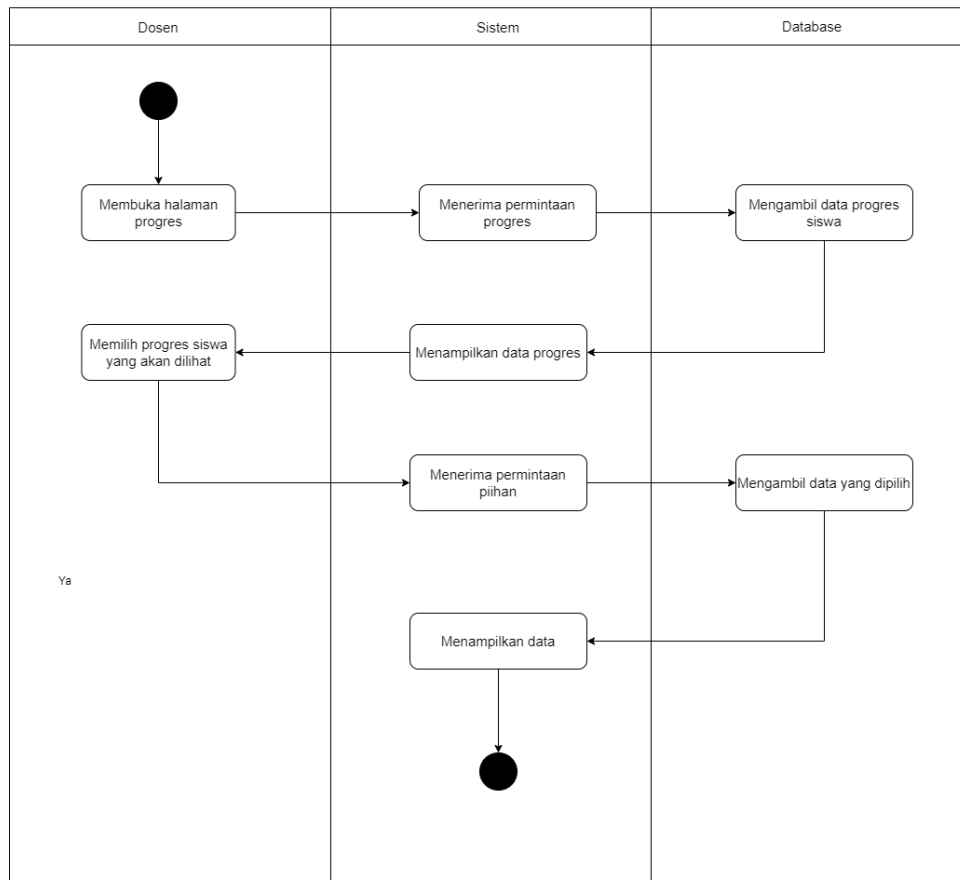
4. Hapus Topik atau Materi



Gambar 3.8 *Activity diagram* hapus topik atau materi

Ini merupakan *activity diagram* untuk menghapus topik atau materi. Pada *activity* ini dosen dapat melakukan penghapusan pada topik atau materi yang telah dibuat sebelumnya. *Activity* ini dimulai dengan dosen masuk ke halaman topik atau materi, kemudian sistem akan menampilkan daftar topik atau materi yang ada, dosen dapat memilih topik atau materi yang akan dihapus dengan menekan *button* hapus pada topik atau materi yang dipilih, data yang telah dipilih akan dihapus pada *database*.

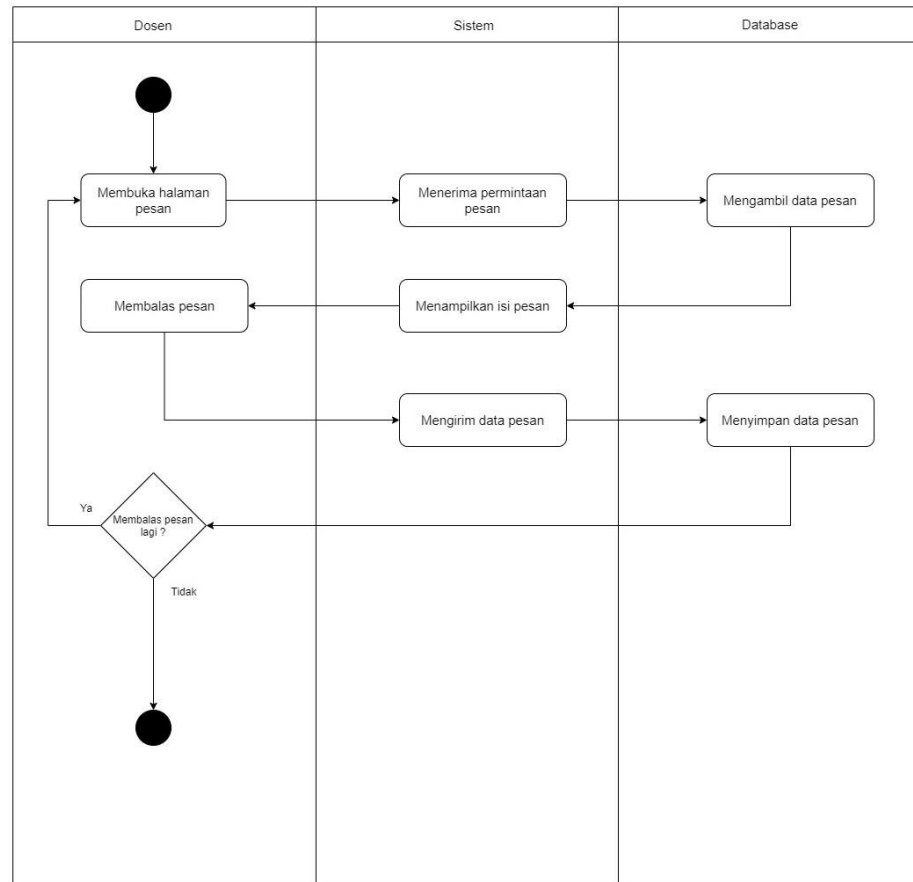
5. Lihat *Progress* Mahasiswa



Gambar 3.9 *Activity diagram* melihat *progress* mahasiswa

Ini merupakan *activity diagram* untuk melihat *progress* mahasiswa. Pada *activity* ini dosen dapat melakukan pemantauan terhadap belajar mahasiswa. *Activity* ini dimulai dengan dosen masuk ke halaman *progress* mahasiswa, kemudian sistem akan menampilkan data *progress* mahasiswa yang diambil dari *database*. *Progress* belajar mahasiswa yang akan ditampilkan, yaitu nama mahasiswa, total waktu belajar, isian dari *subgoal*, isian dari SRL Palette dan pesan dari mahasiswa.

6. Membalas Pesan

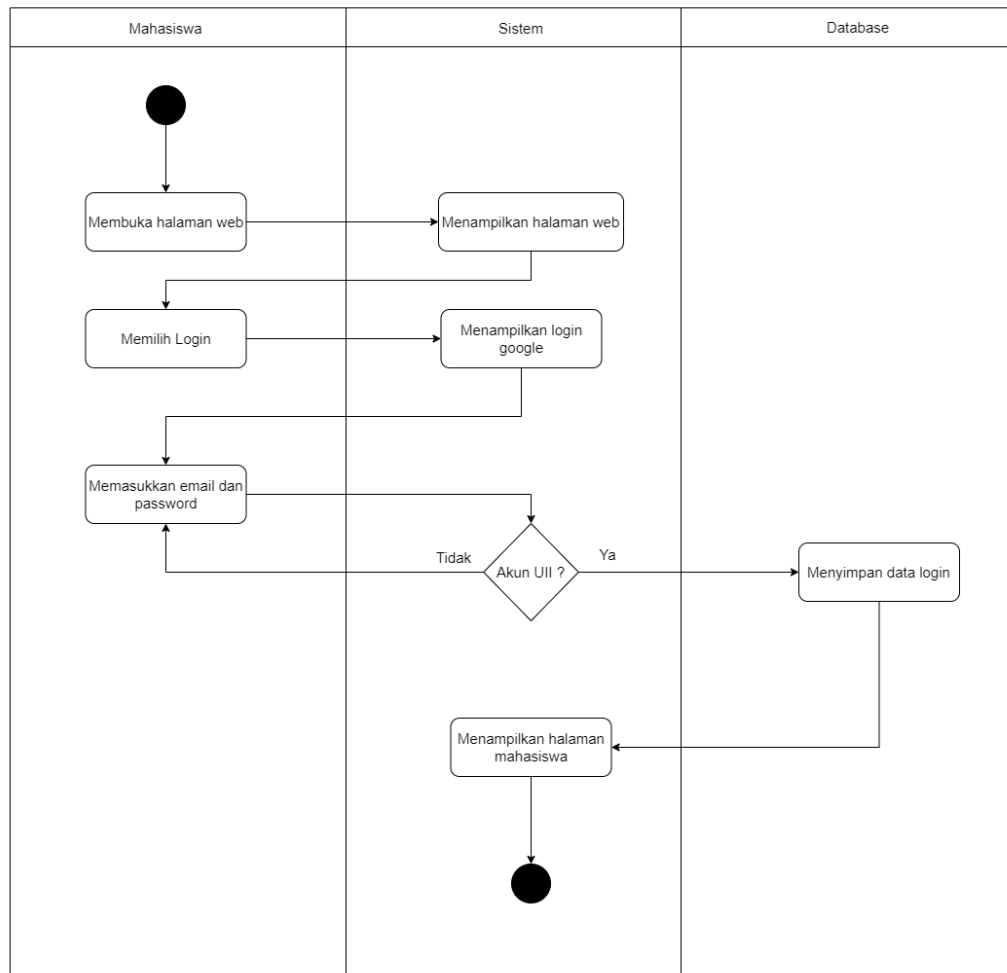


Gambar 3.10 *Activity diagram* membalas pesan

Ini merupakan *activity diagram* untuk membalas pesan. Pada *activity* ini dosen dapat menjawab pertanyaan yang diberikan mahasiswa di setiap materinya. *Activity* ini dimulai dengan dosen membuka halaman pesan, kemudian sistem akan menampilkan isi pesan pada *form chat*, kemudian dosen dapat membalas pesan dengan mengisi *form* tersebut dan mengirimnya ke mahasiswa. Dosen dapat mengisi pesan lagi atau sudah akhiri.

b. *Activity Diagram Siswa*

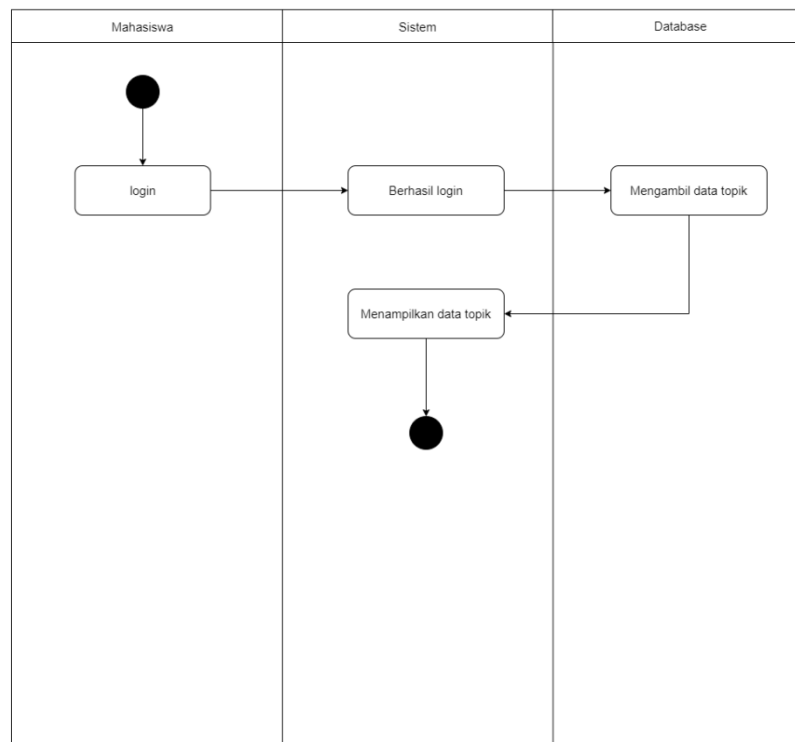
1. *Login*



Gambar 3.11 *Activity Diagram Login*

Ini merupakan *activity diagram login*. *Activity* ini harus dilakukan oleh mahasiswa untuk dapat masuk ke halaman mahasiswa. Pertama mahasiswa membuka alamat web kemudian memilih *login*, setelah memilih *login* masukkan alamat *email* UII, kemudian akan diarahkan ke *login gateway* UII, kemudian mengisi *username* dan *password* UII. Jika yang dimasukkan bukan akun UII maka tidak bisa masuk ke *website* dan akan mengulangi untuk mengisi *email* lagi.

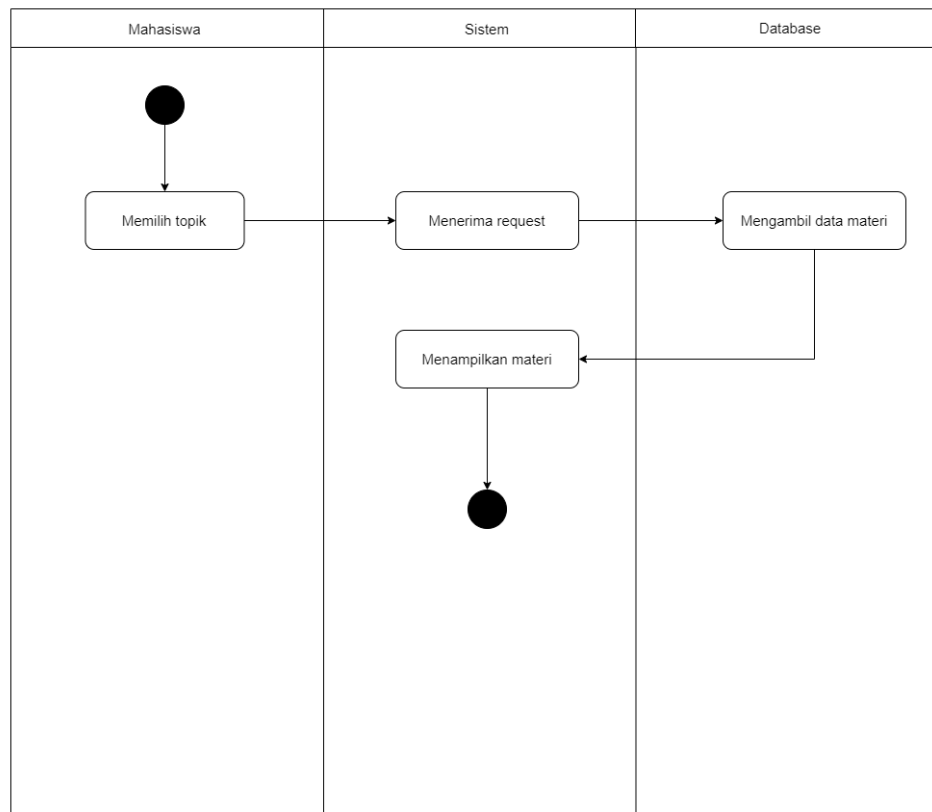
2. Melihat Topik



Gambar 3.12 *Activity diagram* melihat topik

Ini merupakan *activity diagram* melihat topik. Pada *activity* ini mahasiswa dapat melihat daftar topik yang telah ditambahkan dosen ke dalam *website*. *Activity* ini dimulai dengan mahasiswa *login* terlebih dahulu kemudian sistem akan memeriksa *login*, jika *login* berhasil sistem akan menampilkan daftar topik. Data topik yang ditampilkan, yaitu nama topik dan tujuan dari belajar topik tersebut.

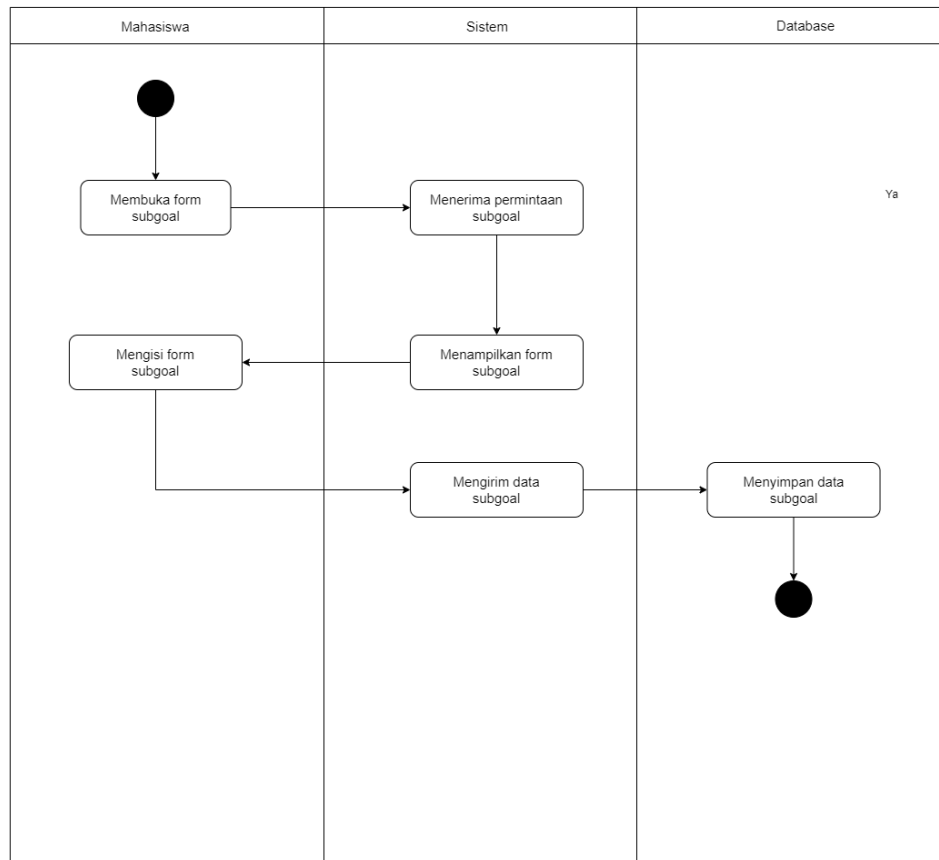
3. Melihat Materi



Gambar 3.13 *Activity diagram* melihat materi

Ini merupakan *activity* untuk melihat materi. Pada *activity* ini mahasiswa dapat melihat daftar materi yang telah ditambahkan dosen disetiap topiknya. *Activity* ini dimulai dengan mahasiswa memilih topik terlebih dahulu, setelah memilih topik maka sistem akan mengambil daftar materi yang merupakan bagian dari topik tersebut dari *database*, untuk ditampilkan ke halaman materi. Data yang akan ditampilkan, yaitu video atau gambar dan judul materi.

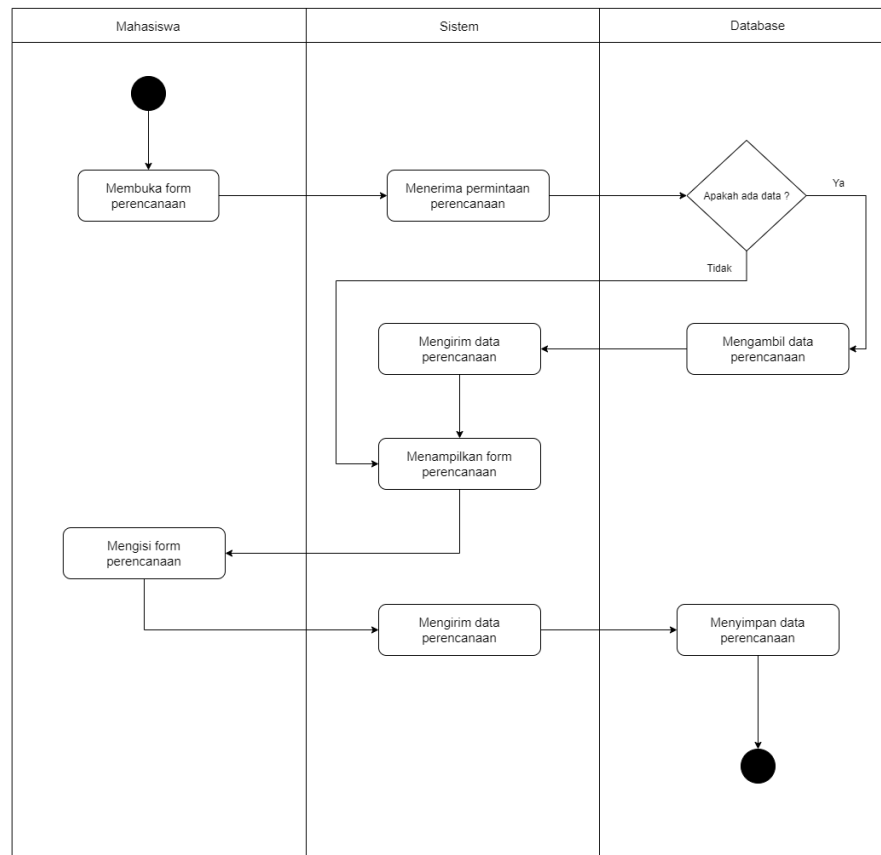
4. Menambah *Subgoal*



Gambar 3.14 *Activity diagram* menambah *subgoal*

Ini merupakan *activity diagram* menambah *subgoal*. Pada *activity* ini mahasiswa akan mengisi tujuan dari belajarnya di setiap materi yang dipelajari. *Activity* ini dimulai dengan mahasiswa membuka *form subgoal*, kemudian sistem akan menampilkan *form*, kemudian mahasiswa dapat melakukan pengisian *subgoal* pada *form* tersebut, setelah mengisi *form* mahasiswa dapat mengirim *subgoal* untuk disimpan ke dalam *database*.

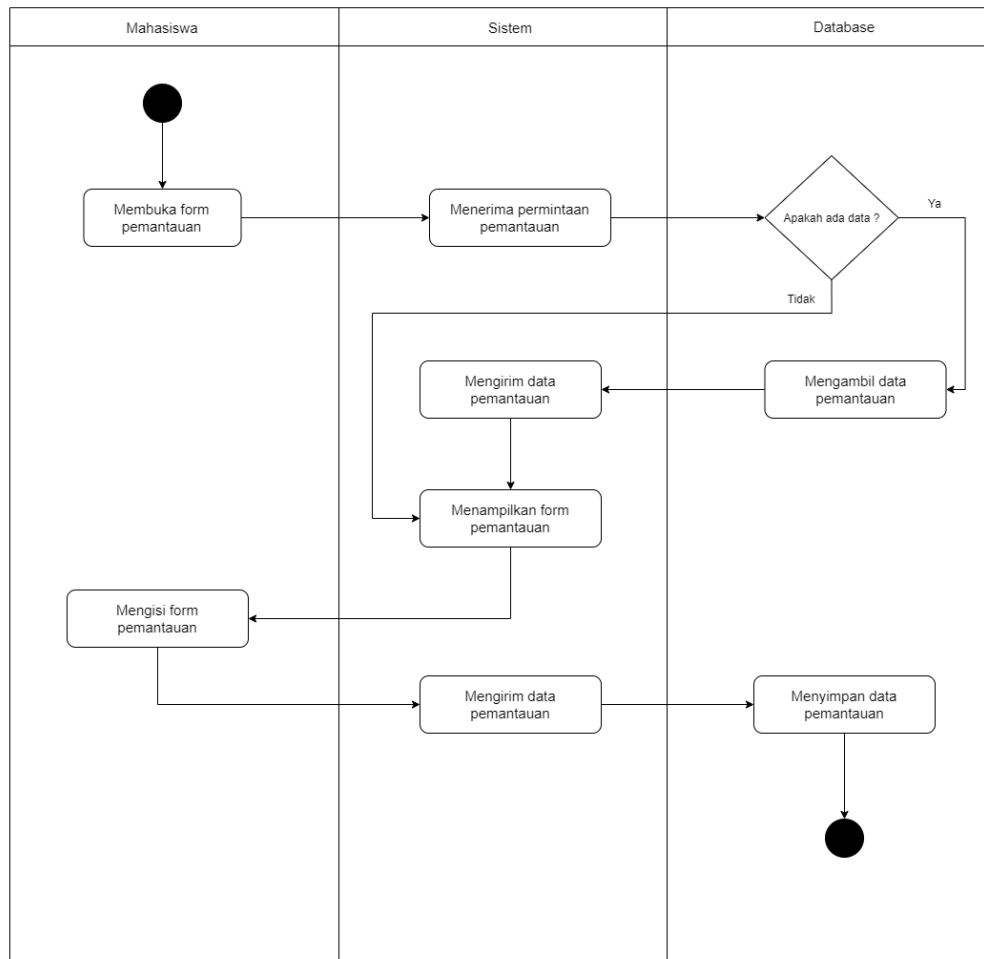
5. Menambah Perencanaan



Gambar 3.15 *Activity diagram* menambah perencanaan

Ini merupakan *activity diagram* menambah perencanaan. Pada *activity* ini mahasiswa akan mengisi mengenai *Plan my learning by* yang memiliki dua isian, yaitu *Stating my understanding about the task* dan *What I have known about the topic*. *Activity* ini dimulai dengan mahasiswa membuka *form* pada bagian *plan*, sistem akan memeriksa pada *database* apakah ada data atau tidak, jika ada maka data tersebut akan diambil dan ditampilkan pada *form*, jika tidak ada maka sistem hanya akan menampilkan *form*, mahasiswa dapat langsung mengisi *form* tersebut, setelah mengisi data sistem akan menyimpan ke *database*.

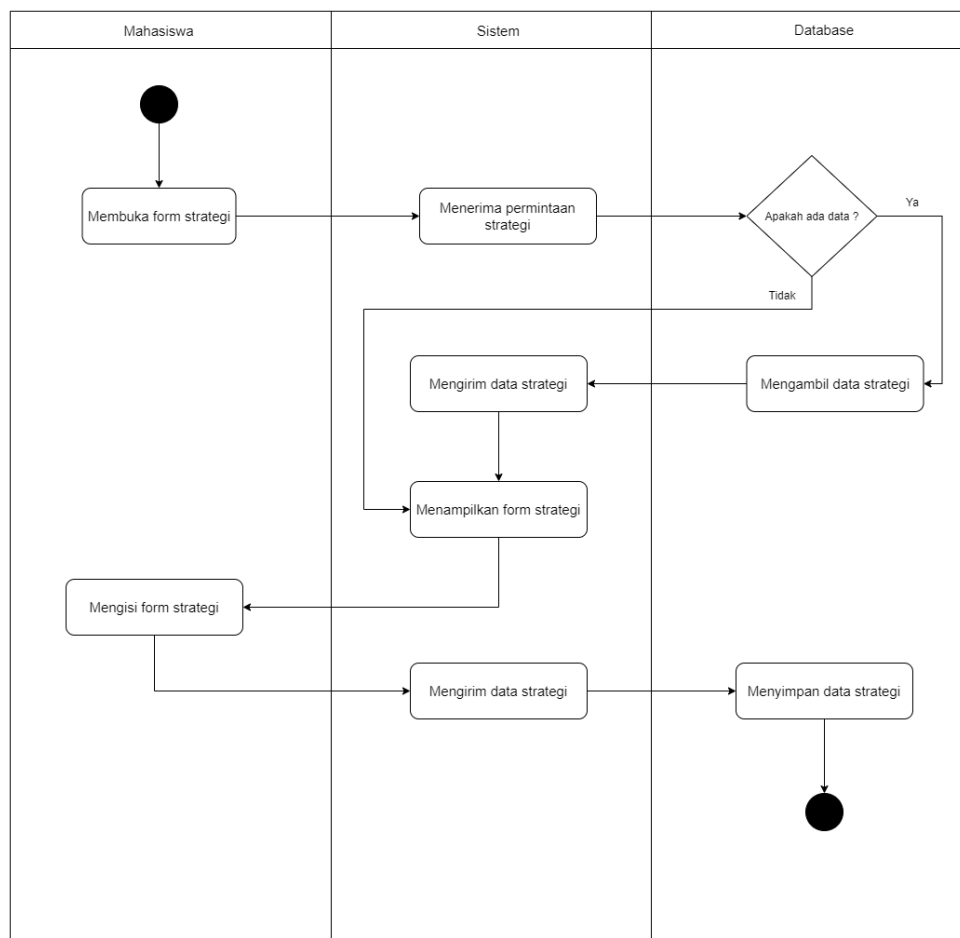
6. Menambah Pemantauan



Gambar 3.16 *Activity diagram* menambah pemantauan

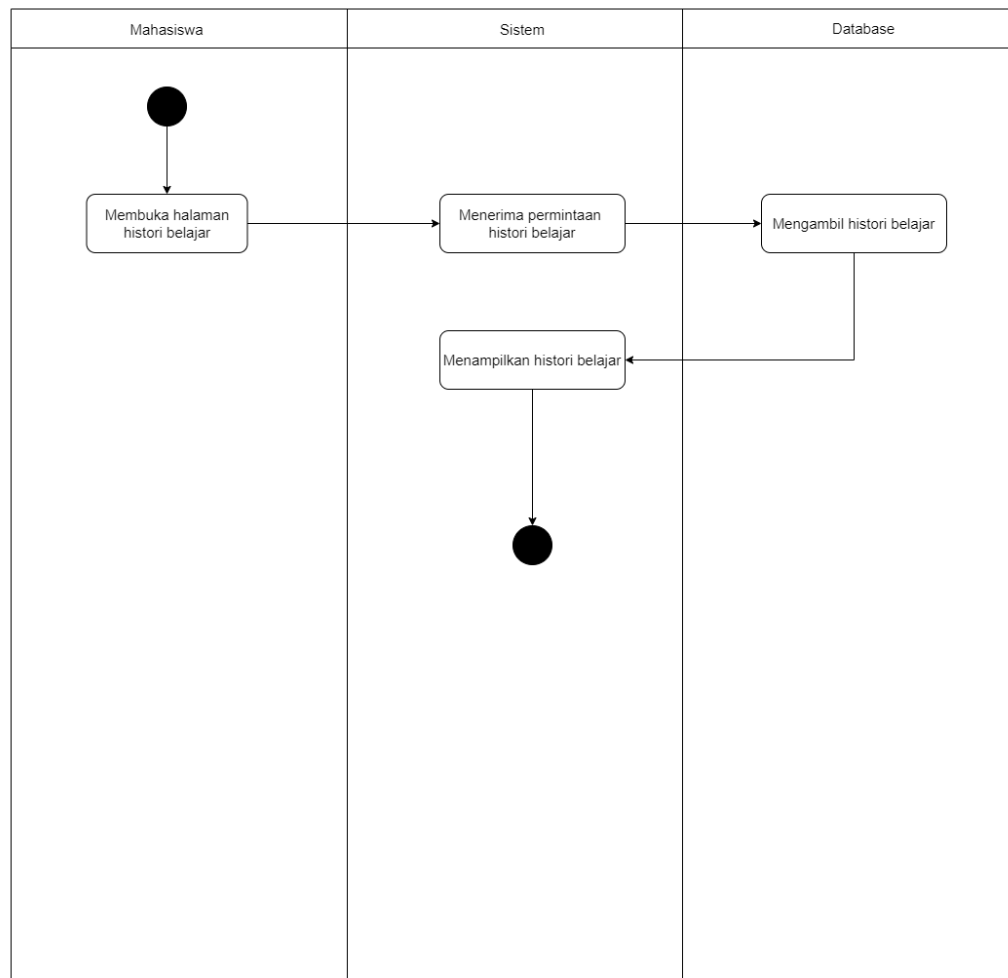
Ini merupakan *activity diagram* menambah pemantauan. Pada *activity* ini mahasiswa akan mengisi mengenai *Monitor my learning by* yang memiliki tiga isian, yaitu *Assesing how well I understand this*, *valuating how well I already know about this content* dan *Evaluating how well the content matches my current subgoals*. *Activity* ini dimulai dengan mahasiswa membuka *form* pada *monitoring*, sistem akan memeriksa pada *database* apakah ada data atau tidak, jika ada maka data tersebut akan diambil dan ditampilkan pada *form*, jika tidak ada maka sistem hanya akan menampilkan *form*, mahasiswa akan mengisi pada *form*, setelah mengisi data sistem akan menyimpan ke *database*.

7. Menambah Strategi



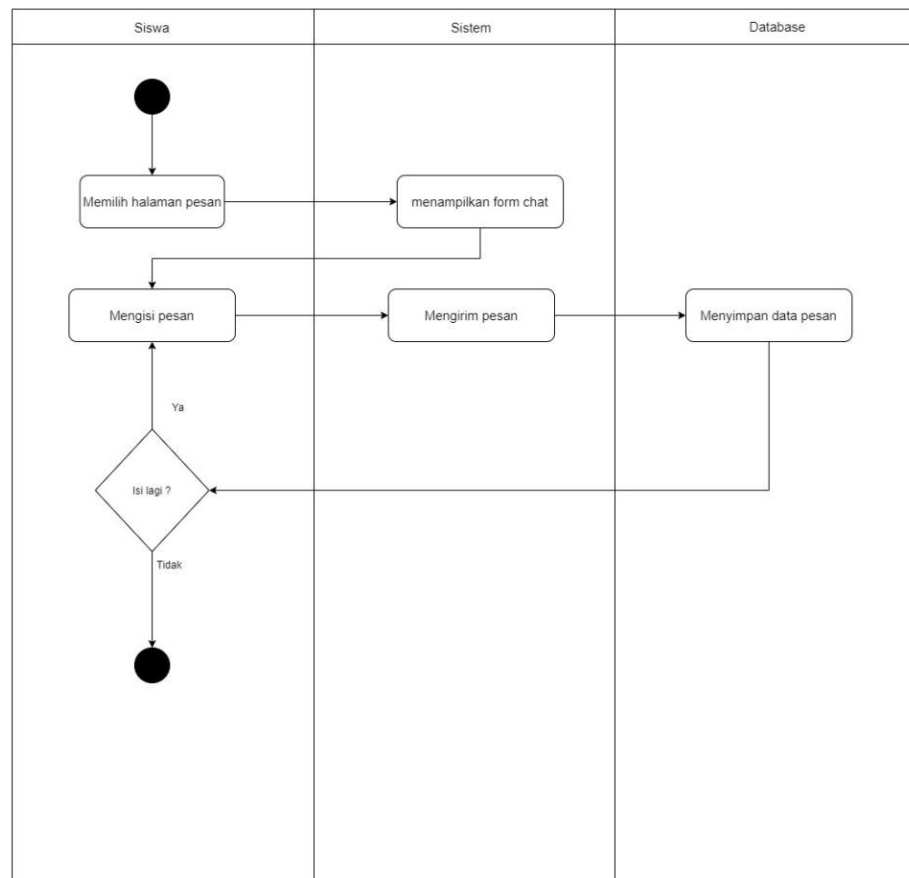
Gambar 3.17 *Activity diagram* menambah strategi

Ini merupakan *activity diagram* menambah strategi. Pada *activity* ini mahasiswa akan mengisi mengenai *Apply a learning strategy* yang memiliki empat isian, yaitu *Take Notes*, *Summarize*, *Make an inferences* dan *Asking for help*. *Activity* ini dimulai dengan mahasiswa membuka *form* isian pada *monitoring*, sistem akan memeriksa pada *database* apakah ada data atau tidak, jika ada maka data tersebut akan diambil dan ditampilkan pada *form*, jika tidak ada maka sistem hanya akan menampilkan *form*, mahasiswa dapat melakukan pengisian pada *form* tersebut, setelah mengisi data sistem akan menyimpan ke *database*.

8. Melihat *History* BelajarGambar 3.18 *Activity diagram* melihat *history* belajar

Ini merupakan *activity diagram* melihat *history*. Pada *activity* ini mahasiswa dapat melihat sendiri riwayat belajarnya. *Activity* ini dimulai dengan mahasiswa membuka halaman *history*, kemudian sistem akan menampilkan semua data *history* belajarnya dari *database*. Data yang akan ditampilkan, yaitu nama materi, total waktu belajar, isian *subgoal* dan isian SRL Palette.

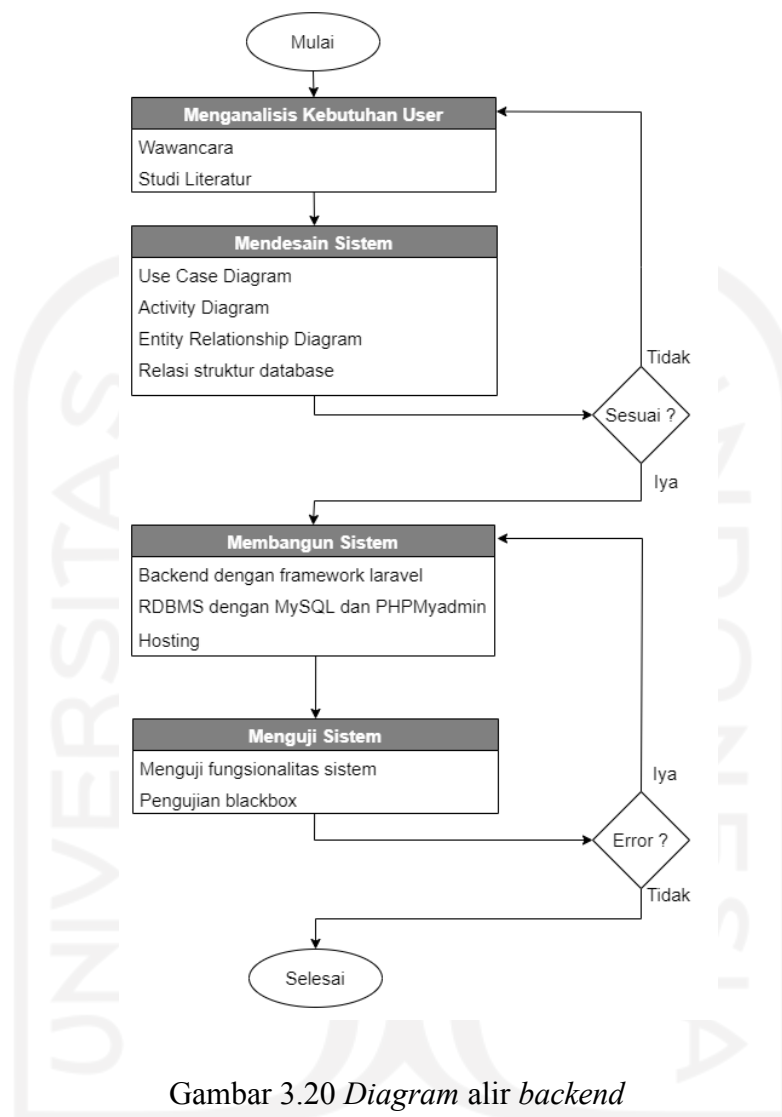
9. Mengirim Pesan



Gambar 3.19 *Activity diagram* mengirim pesan

Ini merupakan *activity diagram* mengirim pesan. Pada *activity* ini mahasiswa dapat mengirim pesan kepada dosen untuk mengajukan pertanyaan terkait materi yang sedang dipelajarinya. *Activity* ini dimulai dengan mahasiswa membuka halaman pesan pada materi tersebut, kemudian sistem akan menampilkan *form*, selanjutnya mahasiswa dapat mengisi *form* pesan dan mengirimnya. Mahasiswa dapat mengisi pesan lagi atau sudah akhiri.

3.3.5 Diagram Alir Backend



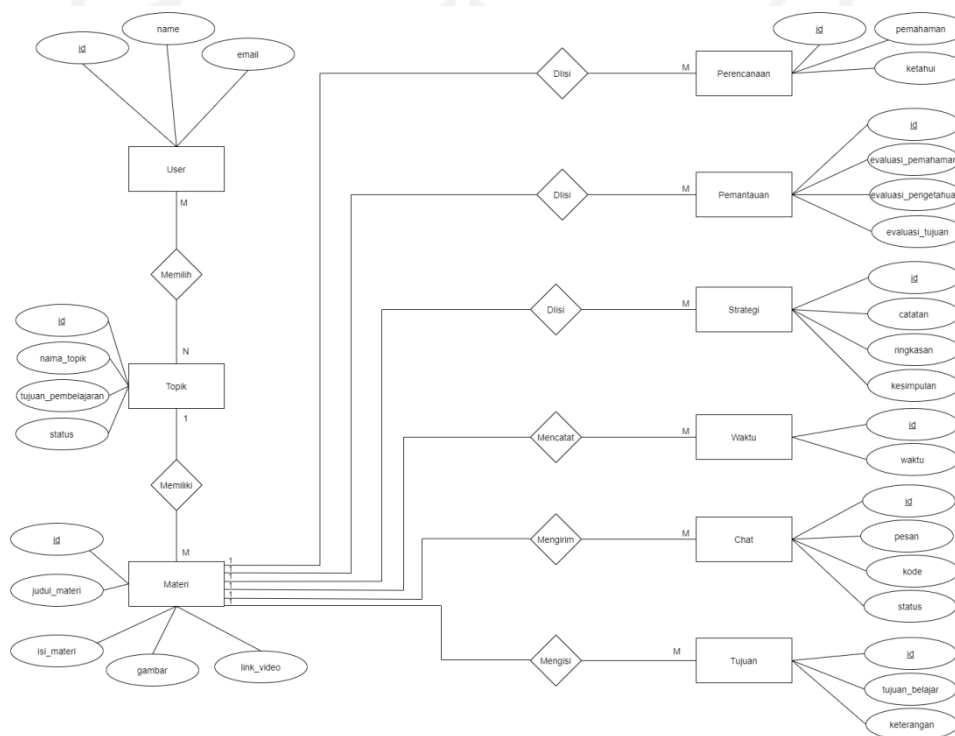
Gambar 3.20 Diagram alir backend

Pada pengembangan *website* dibagi menjadi tiga bagian, yaitu bagian UI/UX, *frontend* dan *backend*. Pada bagian *backend* dimulai dengan menganalisis kebutuhan dari *client*. Analisis kebutuhan terdiri dari studi literatur dan wawancara dengan *client* melalui aplikasi zoom meeting. Setelah kebutuhan-kebutuhan dari *client* didapatkan maka selanjutnya adalah mendesain sistem yang terdiri dari *use case diagram*, *activity diagram*, *Entity Relationship Diagram*, dan relasi struktur *database*. Pada mendesain sistem ini, jika ada bagian yang masih kurang maka akan melakukan studi literatur atau wawancara lagi, tetapi jika sudah sesuai maka akan ketahap selanjutnya, yaitu membangun sistem. Pada tahap ini terdiri dari membuat *backend* dengan *framework* laravel dan membuat *database* dengan MySQL di

phpMyadmin. Jika web sudah selesai, maka akan dilakukukan *hosting*. Setelah sistem jadi maka tahap selanjutnya adalah menguji sistem, pengujian sistem dibagian *backend* menggunakan pengujian *black box testing*, pengujian ini akan menguji sistem dari sisi fungsionalitas web. Jika saat pengujian mengalami masalah atau *error* maka akan menuju ketahap membangun sistem lagi untuk dilakukannya perbaikan.

3.3.6 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah rancangan atau gambaran dalam pembuatan *database* untuk menggambarkan relasi pada entitas-entitas yang ada. Di bawah Ini merupakan rancangan ERD pada aplikasi web *online self-regulated learning* (ON-SRUII).



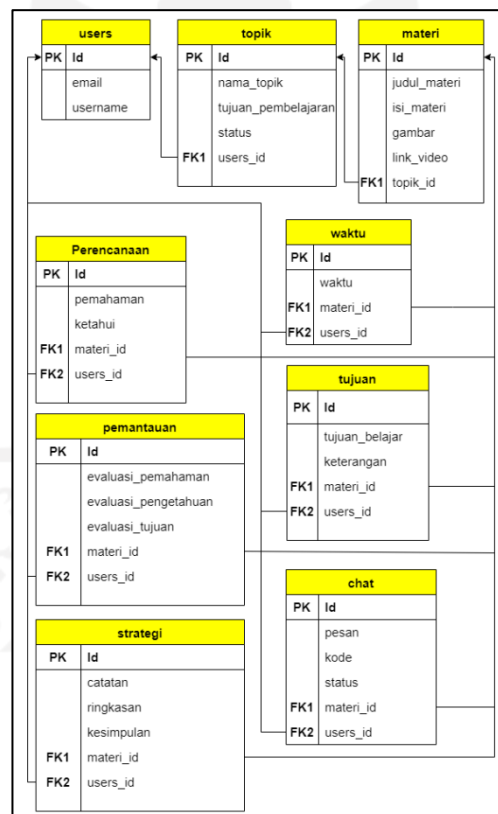
Gambar 3.21 *Entity Relationship Diagram* (ERD)

Entitas yang ada pada ERD ON-SRUII sebagai berikut:

- Entitas *User*
Atributnya : *id, name, email*
- Entitas *Topik*
Atributnya : *id, nama_topik, tujuan_pembelajaran, status*
- Entitas *Materi*
Atributnya : *id, judul_materi, isi_materi, gambar, link_video*

- d. Entitas Perencanaan
Atributnya : id, pemahaman, ketahui
- e. Entitas Pemantauan
Atributnya : id, evaluasi_pemahaman, evaluasi_pengetahuan, evaluasi_tujuan
- f. Entitas Strategi
Atributnya : id, catatan, ringkasan, kesimpulan
- g. Entitas Waktu
Atributnya : id, waktu
- h. Entitas *Chat*
Atributnya : id, pesan, kode, status
- i. Entitas Tujuan
Atributnya : id, tujuan_belajar, keterangan

3.3.7 Relasi Struktur Database



Gambar 3.22 Relasi struktur database

Relasi struktur *database* menggambarkan hubungan antar tabel dalam *database* dengan *primary key* dan *foreign key*. PK menyatakan *primary key* digunakan sebagai identitas dari suatu baris pada tabel *database* dan FK menyatakan *foreign key* sebagai pengenal unik untuk menghubungkan dua tabel atau lebih. Pada tabel-tabel tersebut akan digunakan untuk menyimpan berbagai data *input* dari *user* maupun data hasil olahan.

3.4 Pengerjaan Pengembangan *website*

Dalam pengerjaan pengembangan web dibagi menjadi tiga bagian, yaitu UI/UX, *frontend* dan *backend*. UI/UX akan membuat desain tampilan web. *Frontend* akan membuat kode untuk tampilan berdasarkan desain yang dibuat oleh UI/UX. *Backend* akan membuat kode untuk menangani alur web dan merancang *database*. Tentunya agar pengembangan pada *website* dapat berjalan dengan lancar dan selesai tepat waktu maka diperlukan komunikasi antar bagian dan memiliki tujuan yang sama. Komunikasi dalam pengembangan sistem sangatlah penting agar tidak terjadi salah persepsi dan tidak ada perbedaan dalam memahami disetiap bagian pada *website*. Komunikasi yang digunakan adalah dengan menggunakan whatsapp dan zoom meeting. Whatsapp digunakan untuk berdiskusi, memberikan informasi atau ketika anggota tim akan bertanya sesuatu, sedangkan aplikasi zoom meeting digunakan untuk presentasi dan menjelaskan bagian yang sudah jadi kepada anggota tim lain, selain itu zoom meeting juga digunakan untuk berdiskusi ketika ada anggota tim yang sedang menemui hambatan dalam pengerjaannya. Untuk pengiriman *file* yang berkaitan dengan pengembangan web menggunakan google drive.

Pengerjaan bagian *backend* bisa langsung dikerjakan tidak perlu menunggu UI/UX atau *frontend* selesai. Pengerjaan ini berdasarkan hasil dari analisis, ketika UI/UX jadi baru disesuaikan. Setelah *install* laravel, kemudian membuat *database*. Pada laravel disesuaikan dibagian *.env* dan membuat kode awal untuk *route* dan *controllers*, kemudian menunggu UI/UX. Setelah UI/UX selesai maka melakukan zoom meeting untuk UI/UX menjelaskan tentang tampilannya apakah sudah sesuai atau belum, jika belum sesuai atau ada perubahan maka UI/UX akan melakukan perbaikan jika tidak ada maka dilanjutkan pembuatan bagian *frontend* dan *backend*. *Frontend* mengerjakan bagiannya, sedangkan pada bagian *backend* akan menyelesaikan kodenya di *controller* dan *route*, selain itu juga melakukan perbaikan pada *database* jika ada yang belum sesuai. Kemudian menunggu tampilan yang sedang dikerjakan oleh *frontend* ketika ada bagian tampilan yang sudah selesai maka *frontend* akan mengirimkan *file* tampilan lewat google drive. Pada bagian *backend* akan menempatkan *file*

tersebut pada *folder view*, meletakkan *css* pada bagian *public* dan menyesuaikan kode *controllers* dan *route* agar data dapat ditampilkan dan tampilan dapat berjalan sesuai dengan rancangan. Pengerjaannya begitu sampai semua bagian tampilan selesai ketika semua tampilan sudah selesai digabungkan menjadi satu kesatuan, selanjutnya adalah melakukan *zoom meeting* bersama anggota tim antara *UI/UX*, *frontend* dan *backend*, melakukan pengecekan *website* apakah sudah sesuai atau belum, mengecek alur jalannya *website* dan melakukan pengecekan kemungkinan adanya *error* pada *website*. Jika *website* sudah sesuai dan tidak ada *error* maka dilanjutkan *zoom meeting* bersama *client* untuk memastikan semua sudah sesuai.

3.5 Desain Pengujian

Pengujian yang dilakukan adalah dengan menggunakan *black box testing*. *black box* merupakan pengujian yang akan digunakan untuk memeriksa semua fungsi pada perangkat lunak yang telah dibuat, apakah sudah sesuai atau belum terhadap kebutuhan fungsional yang telah dibuat sebelumnya (Yauma et al., 2020). Pengujian dengan *black box* digunakan untuk memastikan semua fungsi dapat berjalan sesuai dengan kebutuhan fungsional perangkat lunak. Pengujian ini dilakukan dengan cara memberikan kasus uji kepada *user*. Dari kasus uji tersebut akan mendapatkan hasil, apakah fungsi-fungsi pada sistem sudah berjalan sesuai dengan kebutuhan atau belum (Susanto & Ayu, 2017). Pengujian ini akan menguji pada bagian fungsionalitas dari *website* ON-SRUII. Pengujian akan dilakukan pada tampilan dosen dan tampilan mahasiswa. Ada beberapa kriteria untuk pengujian, untuk pengujian pada bagian fungsionalitas dosen kriterianya adalah memiliki *email* UII, pengujian dari dosen bisa juga dari mahasiswa, untuk dosen berasal dari jurusan Pendidikan Bahasa Inggris, minimal pengujian berjumlah 3 orang dan pengujian memiliki materi berupa video atau gambar yang akan ditampilkan, untuk pengujian pada fungsionalitas mahasiswa kriterianya adalah memiliki *email* UII, pengujian berasal dari mahasiswa jurusan Pendidikan Bahasa Inggris, pengujian berada pada semester 5 dan sedang belajar mata kuliah *Research Instrument Development and Analysis*, minimal pengujian berjumlah 10 orang dan pengujian memiliki koneksi internet.

Berikut adalah desain pengujian pada *website* ON-SRUII:

- a. Desain pengujian *black box testing* untuk dosen

Berikut adalah desain pertanyaan untuk menguji fungsionalitas dari *user* dosen:

Tabel 3.7 Desain pengujian *black box user* dosen

Skenario	Hasil yang seharusnya	Hasil	
		Berhasil	Gagal
Membuka www.onsruui.com	Menampilkan tampilan beranda.		
Masuk ke halaman dosen	Hanya bisa <i>email</i> UII.		
Mengklik “ <i>student progress</i> ”	Menampilkan data dari mahasiswa, yaitu nama, nim, waktu, <i>subgoal</i> , <i>chat</i> dan SRL Palette.		
Menambahkan Topik	Berhasil menambahkan topik dan menampilkannya di <i>list</i> topik.		
Menghapus Topik	Berhasil menghapus topik terpilih dari daftar .		
Memperbarui Topik	Berhasil memperbarui data pada topik terpilih.		
Menambahkan Materi	Berhasil menambahkan materi dan menampilkannya di <i>list</i> materi.		
Menghapus Materi	Berhasil menghapus materi terpilih dari daftar.		
Memperbarui Materi	Berhasil memperbarui data pada materi terpilih.		
Mengirim Pesan	Pesan berhasil dikirim dan masuk ke riwayat pesan.		
Mengklik “ <i>details SRL Palette</i> ”	Menampilkan isian SRL Palette dari mahasiswa.		

b. Desain pengujian *black box testing* untuk mahasiswa

Berikut adalah desain pertanyaan untuk menguji fungsionalitas sistem dari sisi *user* mahasiswa:

Tabel 3.8 Desain pengujian *black box user* mahasiswa

Skenario	Hasil yang seharusnya	Hasil	
		Berhasil	Gagal
Membuka www.onsruui.com	Menampilkan tampilan beranda.		
Masuk ke halaman mahasiswa	Hanya bisa <i>email</i> UII.		
Memilih topik pada halaman utama mahasiswa	Menampilkan daftar materi.		
Mengklik " <i>history</i> "	Menampilkan semua riwayat belajar dari nama materi, waktu, <i>subgoals</i> dan SRL Palette.		
Mengklik " <i>complete</i> "	Akan menghapus daftar di <i>subgoals</i> , karena <i>subgoals</i> tersebut telah selesai dan <i>subgoals</i> tersebut akan masuk ke <i>history</i> .		
<i>Add subgoals</i>	Berhasil menambahkan <i>subgoals</i> baru .		
<i>Add Stating my understanding about the task</i>	Berhasil menambahkan data pada <i>form</i> dan <i>form</i> akan menampilkan data jika pernah mengisi.		
<i>Add What I have known about the topic</i>	Berhasil menambahkan data pada <i>form</i> dan <i>form</i> akan menampilkan data jika pernah mengisi.		
<i>Add Assesing how well I understand this</i>	Berhasil menambahkan data pada <i>form</i> dan <i>form</i> akan menampilkan data jika pernah mengisi.		

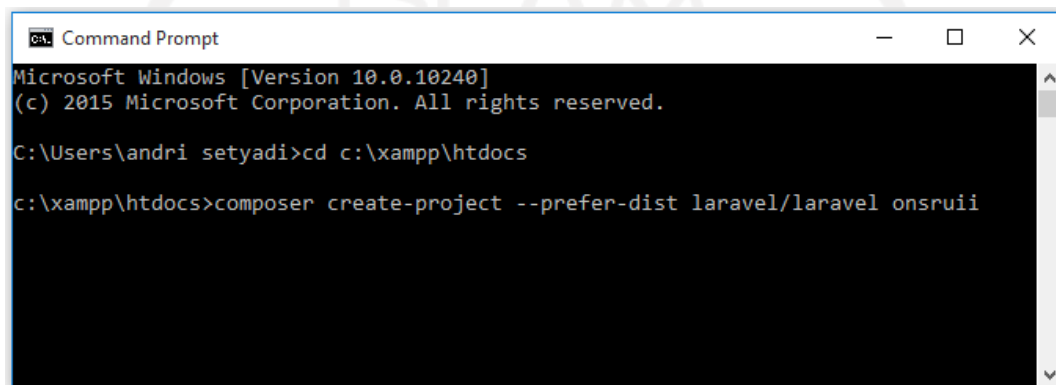
<i>Add Evaluating how well I already know about this content</i>	Berhasil menambahkan data pada <i>form</i> dan <i>form</i> akan menampilkan data jika pernah mengisi.		
<i>Add Evaluating how well the content matches my current subgoals</i>	Berhasil menambahkan data pada <i>form</i> dan <i>form</i> akan menampilkan data jika pernah mengisi.		
<i>Add Take Notes</i>	Berhasil menambahkan data pada <i>form</i> dan <i>form</i> akan menampilkan data jika pernah mengisi.		
<i>Add Summarize</i>	Berhasil menambahkan data pada <i>form</i> dan <i>form</i> akan menampilkan data jika pernah mengisi.		
<i>Add Make an inferences</i>	Berhasil menambahkan data pada <i>form</i> dan <i>form</i> akan menampilkan data jika pernah mengisi.		
<i>Add Chat</i>	Berhasil mengirim <i>chat</i> dan menampilkan riwayat <i>chat</i> .		

BAB IV

HASIL DAN PEMBAHASAN

4.1 *Develop Backend Website*

Develop backend ini dilakukan setelah proses analisis, perancangan dan desain sudah selesai. Pada *develop web* akan membuat fungsi-fungsi di *framework* laravel dan membuat *database* di phpMyAdmin untuk *website* ON-SRUII.



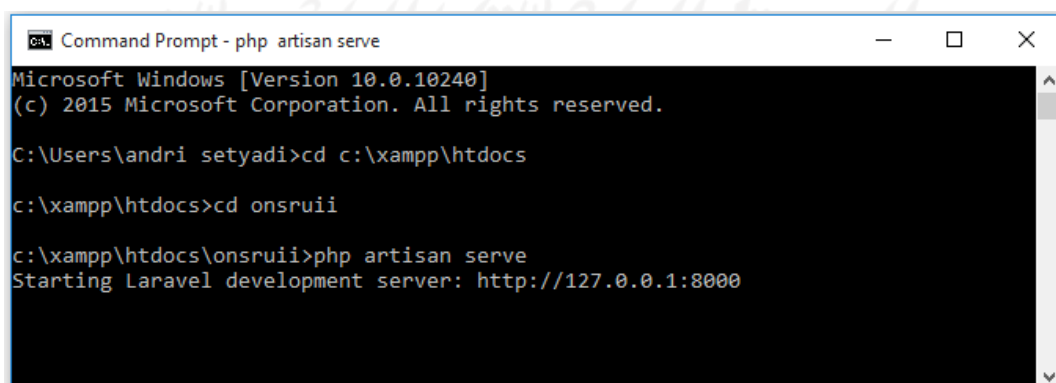
```
ca. Command Prompt
Microsoft Windows [Version 10.0.10240]
(c) 2015 Microsoft Corporation. All rights reserved.

C:\Users\andri setyadi>cd c:\xampp\htdocs

c:\xampp\htdocs>composer create-project --prefer-dist laravel/laravel onsruii
```

Gambar 4.1 *Install laravel*

Sebelum *install* laravel arahkan terlebih dahulu ke *folder* *htdocs* pada *command prompt*, dengan cara *cd* (fungsi untuk mengubah lokasi) nama *drive*, kemudian nama *folder* *xampp* dan *folder* *htdocs*, kemudian *enter* maka pada *command prompt* tujuan direktori akan berubah. Setelah direktori mengarah ke *htdocs* dilanjutkan dengan *install* laravel dengan cara *composer create-project --prefer-dist laravel/laravel onsruii*, seperti pada Gambar 4.1, *project* ini diberi nama *onsruii*. Proses *install project* laravel memerlukan waktu sekitar 5 menit.



```
ca. Command Prompt - php artisan serve
Microsoft Windows [Version 10.0.10240]
(c) 2015 Microsoft Corporation. All rights reserved.

C:\Users\andri setyadi>cd c:\xampp\htdocs

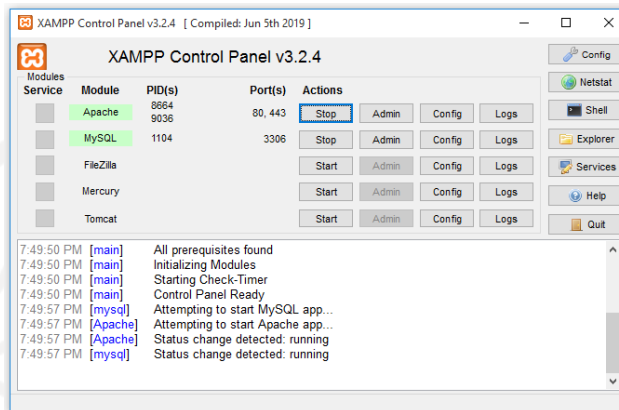
c:\xampp\htdocs>cd onsruii

c:\xampp\htdocs\onsruii>php artisan serve
Starting Laravel development server: http://127.0.0.1:8000
```

Gambar 4.2 *Start project laravel*

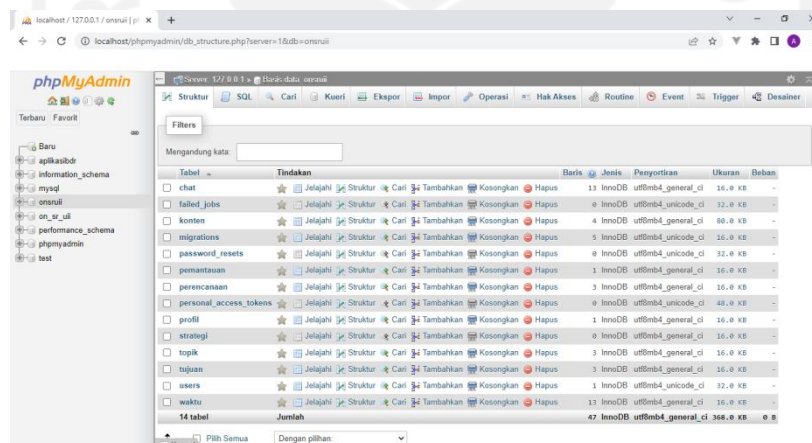
Setelah proses *install* selesai dilanjutkan dengan menjalankan projek tersebut lewat *cmd* dengan cara *c:\xampp\htdocs\onsruii* kemudian masukkan perintah *php artisan serve*.

Setelah *enter* akan muncul satu baris yang berisi alamat *localhost*, alamat tersebut akan digunakan untuk menjalankan *project* laravel onsruii. Pada Gambar 4.2 merupakan alamat dari *localhost*, yaitu *http://127.0.0.1:8000*. Salin alamat tersebut ke *browser* untuk menjalankan web yang sedang dikembangkan dengan laravel.



Gambar 4.3 Tampilan XAMPP

Membuat *database* di phpMyAdmin dengan membuka XAMPP terlebih dahulu, kemudian *start* Apache dan MySQL.



Gambar 4.4 Tabel *database* onsruii

Setelah itu, membuka *http://localhost/phpmyadmin* untuk membuat *database* dan tabel sesuai dengan desain *database* yang telah dibuat, setelah *database* selesai dibuat maka selanjutnya menyesuaikan nama *database* di phpMyAdmin dengan yang ada di *project* laravel.

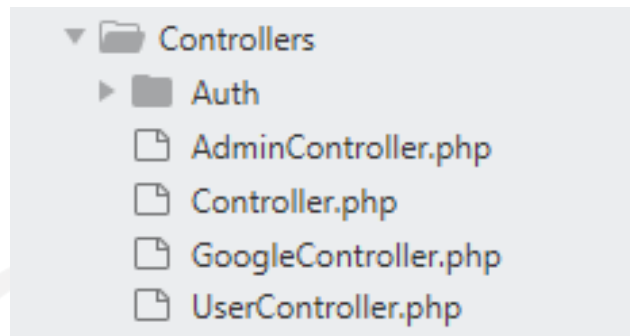
```

9
10 DB_CONNECTION=mysql
11 DB_HOST=127.0.0.1
12 DB_PORT=3306
13 DB_DATABASE=onsruii
14 DB_USERNAME=root
15 DB_PASSWORD=
16

```

Gambar 4.5 Laravel *file .env*

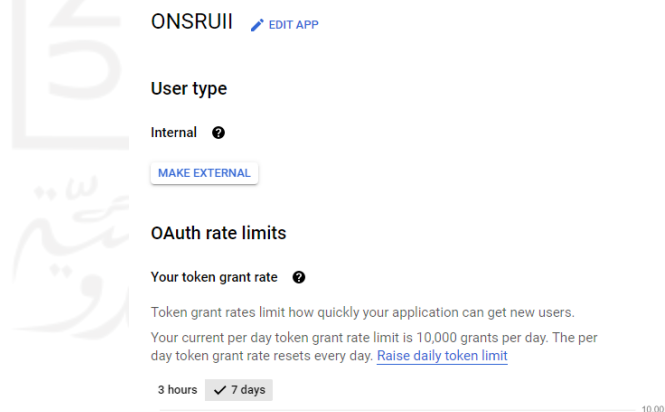
Pada phpMyAdmin *database* diberi nama *onsruii* maka di laravel juga harus disesuaikan pada *file .env* di bagian *DB_DATABASE = onsruii*.



Gambar 4.6 Laravel *folder controllers*

Membuat fungsi-fungsi yang akan digunakan pada *website* untuk mengolah atau menghubungkan antara *view* dengan *database*. Fungsi-fungsi tersebut terdapat pada tiga *controllers*, yaitu pada *AdminController.php* (untuk menyimpan fungsi-fungsi yang akan digunakan pada halaman dosen), *GoogleController.php* (untuk menyimpan fungsi *login* dengan akun UII dan dibagian ini ada fungsi yang digunakan sebagai pembeda antara *user* dosen dan *user* mahasiswa), *UserController.php* (untuk menyimpan fungsi-fungsi yang akan digunakan pada halaman mahasiswa).

Login pada *website* menggunakan akun UII, hanya *user* yang mempunyai akun UII. Penggunaan *login* ini dengan memanfaatkan API google, pertama membuka <https://console.developers.google.com> dan membuat *project* baru didalamnya.



Gambar 4.7 Pengaturan *login*

Setelah itu, melakukan pengaturan dari *app name*, *user type*, dan *user support email*. *App name* adalah nama dari *website*, *user type* adalah jenis *user* yang bisa *login* ke *website* jika memilih internal maka hanya *email* organisasi *uii.ac.id* yang bisa masuk, jika memilih eksternal maka siapapun yang memiliki akun google dapat masuk, dan *user support email*

adalah alamat *email* yang bisa dihubungi pada *website* ini. Membuat *OAuth Client ID* dengan mengisi URIs: *http://onsruii.com/auth/callback*.

Authorized redirect URIs ⓘ

For use with requests from a web server

URIs 1 *
http://onsruii.com/auth/callback

URIs 2 *
http://tugasakhir.onsruii.com/auth/callback

+ ADD URI

Note: It may take 5 minutes to a few hours for settings to take effect

SAVE CANCEL

Gambar 4.8 *Authorized redirect*

Jadi, ini berfungsi untuk mengarahkan *user* setelah berhasil *login* dengan akun google mereka. Langkah selanjutnya akan mendapatkan *Client ID* dan *Client Secret*, seperti pada gambar 4.9.

Client ID	779772984409-8uendllp1jdfqhogn5gkkaa028ktpdi.apps.googleusercontent.com
Client secret	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
Creation date	November 8, 2021 at 1:57:52 PM GMT+7

Gambar 4.9 *Client ID* dan *Client Secret*

Client ID dan *Client Secret* tersebut disalin untuk dimasukkan ke laravel diletakkan di *file.env* yang dihubungkan ke folder *config /services.php*.

```
'google' => [
    'client_id'      => env('GOOGLE_CLIENT_ID'),
    'client_secret' => env('GOOGLE_APP_SECRET'),
    'redirect'      => env('GOOGLE_REDIRECT'),
],
```

Gambar 4.10 *services.php*

```
GOOGLE_CLIENT_ID=719917120297q61uh7dmvojpftbh9qevsjn31i7qgj2o.apps.googleusercontent.com
GOOGLE_APP_SECRET=49139YgXZrZbgojc1XVmXrg2
GOOGLE_REDIRECT=http://127.0.0.1:8000/auth/callback
```

Gambar 4.11 Laravel *file .env*

4.1.1 Routes

Routes ini adalah bagian dari laravel yang berfungsi sebagai penghubung antara fungsi-fungsi di *controller* dengan *views*. Berdasarkan kebutuhan yang telah dirancang maka berikut *route* yang telah dibuat dari *user* dosen dan mahasiswa.

```
Route::get('/tampilanAdmin','AdminController@bagianAdmin');
Route::get('/lihatTopic','AdminController@lihatTopic');
Route::get('/addTopic','AdminController@addTopic');
Route::post('/simpanTopic','AdminController@simpanTopic');
Route::get('/hapusTopic/{id}','AdminController@hapusTopic');
Route::get('/updateTopic/{id}','AdminController@updateTopic');
Route::post('/simpanUpdateTopic','AdminController@simpanUpdateTopic');
Route::get('/lihatMaterial/{id}','AdminController@lihatMaterial');
Route::get('/addMaterial','AdminController@addMaterial');
Route::post('/simpanMaterial','AdminController@simpanMaterial');
Route::get('/updateMaterial/{id}','AdminController@updateMaterial');
Route::post('/simpanUpdateMaterial','AdminController@simpanUpdateMaterial');
Route::get('/hapusMaterial/{id}','AdminController@hapusMaterial');
Route::get('/updateStatus/{id}','AdminController@updateStatus');
```

Gambar 4.12 *Route* untuk *user* dosen

```
//Progres Siswa
Route::get('/lihatDaftarMateri','AdminController@lihatDaftarMateri');
Route::get('/cari','AdminController@cari');
Route::get('/lihatProgresMateri/{id}','AdminController@lihatProgresMateri');
Route::get('/detailsSubgoals/{id}','AdminController@detailsSubgoals');
Route::get('/SRLPalette/{id}','AdminController@SRLPalette');
Route::get('/chat/{id}','AdminController@chat');
Route::post('/simpanPesan','AdminController@simpanPesan');
//Halaman Mahasiswa
Route::get('/listTopic','UserController@listTopic');
Route::get('/listCourse/{id}','UserController@listCourse');
Route::get('/detailCourse/{id}','UserController@detailCourse');
Route::get('/logout','UserController@logout');
//History siswa
Route::get('/historySiswa','UserController@historySiswa');
Route::get('/subgoals/{id}','UserController@subgoals');
Route::get('/srlPaletteSiswa/{id}','UserController@srlPaletteSiswa');
//halaman utama siswa
Route::get('/halamanBelajar','UserController@halamanBelajar');
Route::get('/pilihMateri/{id}','UserController@pilihMateri');
Route::post('/tambahSubgoals','UserController@tambahSubgoals');
Route::get('/komplet','UserController@komplet');
//srl palette Planning
Route::post('/tambahPemahaman','UserController@tambahPemahaman');
Route::post('/tambahPengetahuan','UserController@tambahPengetahuan');
//srl palette Monitoring
Route::post('/tambahEvaPemahaman','UserController@tambahEvaPemahaman');
Route::post('/tambahEvaPengetahuan','UserController@tambahEvaPengetahuan');
Route::post('/tambahEvaTujuan','UserController@tambahEvaTujuan');
//srl palette Strategy
Route::post('/tambahCatatan','UserController@tambahCatatan');
Route::post('/tambahRingkasan','UserController@tambahRingkasan');
Route::post('/tambahKesimpulan','UserController@tambahKesimpulan');
//chat
Route::post('/ kirimChat','UserController@ kirimChat');
```

Gambar 4.13 *Route* untuk *user* mahasiswa

4.1.2 Dosen

a. Menambah topik

Untuk melakukan penambahan topik pada *website*, dosen akan membuka halaman *form* topik pada web yang berada di *view*('dosen/addTopic'). Kemudian melakukan pengisian dari nama topik dan tujuan pembelajaran. Setelah itu, klik *upload* maka akan melalui *route* untuk mengarah ke *AdminController* pada fungsi *simpanTopic*, fungsi tersebut akan memproses data untuk disimpan ke dalam *database* pada tabel *topic*.

```
public function addTopic(Request $request) {
    if (Auth::check()) {
        return view('dosen/addTopic');
    }else{
        return redirect('/logout');
    }
}
public function simpanTopic(Request $request) {
    $messages = [
        'required' => ':attribute is required'
    ];
    $this->validate($request, [
        'topicName' => 'required',
        'studyGoals' => 'required'
    ], $messages);

    DB::table('topik')->insert([
        'nama_topik'=>strip_tags($request->topicName),
        'tujuan_pembelajaran'=>strip_tags($request->studyGoals),
        'users'=>Auth::user()->id,
        'status'=>'active'
    ]);
    return redirect('/lihatTopic');
}
```

Gambar 4.14 Kode sumber untuk menambah topik

b. Menambah materi

Untuk menambahkan materi, *website* menyediakan *form* yang berisi *material name*, *material*, *link* video, dan *image*. Dosen dapat melakukan pengisian antara tiga pilihan, yaitu pertama, *material name*, *material* dan *link* video. Kedua, *material name*, *material* dan *image*. Ketiga, *material name* dan *material*. Setelah Dosen klik *upload* maka akan melalui *route* menuju ke fungsi *simpanMaterial* pada *AdminController*. Pada fungsi ini akan melewati beberapa kondisi untuk menyesuaikan dengan isian dari *form* yang kemudian akan disimpan pada *database* di tabel konten. Untuk gambar akan disimpan di laravel pada *folder* *public/materi*.

```
public function simpanMaterial(Request $request) {
    $messages = [
        'required' => ':attribute is required'
    ];
    $this->validate($request, [
        'materialName' => 'required',
        'material' => 'required'
    ]);
}
```

```

    ], $messages);

    if(empty($request->image)){
        if(empty($request->link)){
            $data = $request->session()->get('topik');
            DB::table('konten')->insert([
                'judul_materi'=>strip_tags($request->materialName),
                'isi_materi'=>$request->material,
                'link_video' =>'',
                'gambar'=>'-',
                'topik'=>$data
            ]);
            $idT = $request->session()->get('topik');
            return redirect('/lihatMaterial/'.$idT);
        }else{
            $data = $request->session()->get('topik');
            DB::table('konten')->insert([
                'judul_materi'=>strip_tags($request->materialName),
                'isi_materi'=>$request->material,
                'link_video' =>strip_tags($request->link),
                'gambar'=>'-',
                'topik'=>$data
            ]);
            $idT = $request->session()->get('topik');
            return redirect('/lihatMaterial/'.$idT);
        }
    }else{
        $file = $request->file('image');
        $nama_file = $file->getClientOriginalName();
        //ini untuk upload ke folder public
        $data = $request->session()->get('topik');
        DB::table('konten')->insert([
            'judul_materi'=>strip_tags($request->materialName),
            'isi_materi'=>$request->material,
            'link_video'=>'',
            'gambar'=>$nama_file,
            'topik'=>$data
        ]);
        $tujuan_upload = 'materi';
        $file->move($tujuan_upload,$nama_file);
        $idT = $request->session()->get('topik');
        return redirect('/lihatMaterial/'.$idT);
    }
}

```

Gambar 4.15 Kode sumber untuk menambah materi

c. Update dan hapus

Untuk melakukan *update* pada topik atau materi dilakukan dengan cara dosen akan memilih terlebih dahulu topik atau materi yang akan di *update*, setelah memilih maka akan ada *id* yang dikirimkan ke *route* yang akan diteruskan ke fungsi. Setelah sampai di fungsi, data pada *database* yang memiliki *id* sama dengan *id* yang ada pada fungsi akan diambil, kemudian akan dikirim halaman *form update*, setelah data diubah klik simpan maka data pada *database* akan berubah.

Untuk melakukan penghapusan topik atau materi dilakukan dengan cara dosen memilih topik atau materi yang akan dihapus, setelah memilih maka akan ada *id* yang dikirm ke *route*

untuk diteruskan ke *fungsi hapusTopic* atau *hapusMaterial*. Setelah sampai di fungsi, data yang memiliki *id* sama akan dihapus pada *database* dengan perintah *where->delete*.

d. Melihat *progress*

Untuk menampilkan *progress* dari mahasiswa akan menggunakan fungsi dari *lihatProgresMateri*. Pada fungsi ini akan menampilkan daftar *history* belajar mahasiswa berdasarkan materi yang dipilih oleh dosen (*\$id*). Penggunaan *join* untuk menampilkan semua data pada kedua tabel jika *id* materi yang dipilih sama.

```
$siswa = DB::table('waktu')
->join('users','waktu.users','=','users.id')
->select('users.name','users.id','users.email','waktu.waktu')
->where('waktu.materi','=','',$id)
->get();

request()->session()->put('idMateri',$id);
return view('dosen/lihatProgresMateri',['siswa'=>$siswa]);
```

Gambar 4.16 Kode sumber untuk melihat *progress*

See Progress					
Name	NIM	SubGoals	Time	Chat	SRL Palette
ANDRI SETYADI	185231	Details	63 minute	Show Chat	Details

Gambar 4.17 Tampilan *progress* untuk dosen

4.1.3 Mahasiswa

a. Login

Untuk pemisahan *login* antara dosen dan mahasiswa dilakukan dengan cara memeriksa pada *email* yang digunakan oleh *user* jika pada *email* ada *@uui* maka akan dinyatakan bahwa *email* tersebut adalah *user* dosen jika tidak ada maka dinyatakan *user* mahasiswa. Gambar 4.19 merupakan kode untuk *user* yang sudah pernah *login*. Kode yang terdapat komentar *#simpan gambar* adalah kode untuk menyimpan gambar akun google. Pada perilsan *website* pertama belum menemukan kode untuk menangani gambar avatar google. Setelah beberapa hari baru menemukan maka kode tersebut untuk menangani *user* yang dahulu pernah *login*.

Untuk *user* yang sudah pernah *login* akan melewati kode pada Gambar 4.19. Pada kode tersebut akan diarahkan langsung ke halaman dosen jika yang masuk *user* dosen dan akan diarahkan ke halaman mahasiswa jika yang masuk *user* mahasiswa.

Untuk *user* baru akan melewati kode pada Gambar 4.18. Pada gambar tersebut ada kode yang digunakan untuk menyimpan data dari *user*, data yang disimpan, yaitu nama, gambar, dan *email*.

```

$newUser = User::create([
    'name' => $oauthUser->name,
    'email' => $oauthUser->email,
    'google_id'=> $oauthUser->id,
    'password' => md5($oauthUser->token)];
//gambar
$gambar = $oauthUser->getAvatar();
DB::table('profil')->insert([
    'email' => $oauthUser->email,
    'foto' => $gambar]);

//Membedakan akun @students dan @uii
Auth::login($newUser);
$str = Auth::user()->email;
$pattern = "@uii/i";
if(preg_match($pattern, $str)){
    return redirect('/tampilanAdmin');
}else{
    return redirect('/listTopic');
}

```

Gambar 4.18 Kode sumber untuk *login* (belum pernah *login*)

```

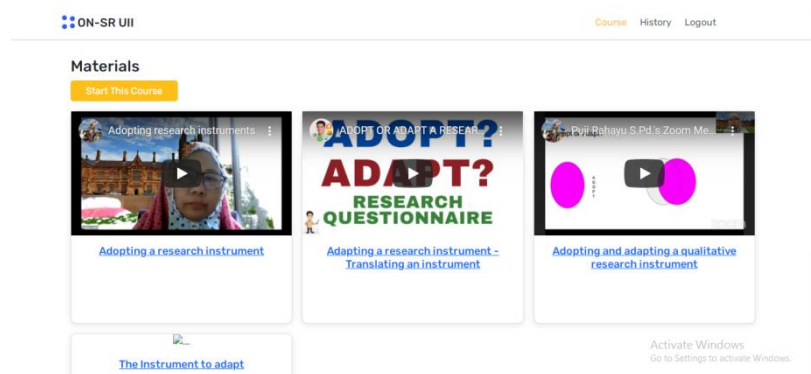
//Akun sudah ada
//Membedakan akun @students dan @uii
Auth::loginUsingId($user->id);
$str = Auth::user()->email;
$pattern = "@uii/i";
$namaA = Auth::user()->name;
    if(preg_match($pattern, $str)){
        return redirect('/tampilanAdmin');
    }else{
        //Login Menjadi Siswa
        //simpan gambar
        $foto = DB::table('profil')->where('email', $str)->first();
        if(is_null($foto)){
            $gambar = $oauthUser->getAvatar();
            DB::table('profil')->insert([
                'email' => $oauthUser->email,
                'foto' => $gambar]);

            return redirect('/listTopic');
        }else{
            //login menjadi siswa
            return redirect('/listTopic');
        }
    }
}

```

Gambar 4.19 Kode sumber untuk *login* (sudah pernah *login*)

b. Halaman materi



Gambar 4.20 Halaman materi

Untuk menampilkan materi pada Gambar 4.20 menggunakan fungsi dari *listCourse*. Pada fungsi tersebut akan menampilkan berbagai materi dari tabel konten berdasarkan topik yang dipilih (materi memiliki *id* yang sama dengan topik) oleh mahasiswa di awal. Hasil pencarian di *database* kemudian akan dikirim ke *view*('mahasiswa/listCourse').

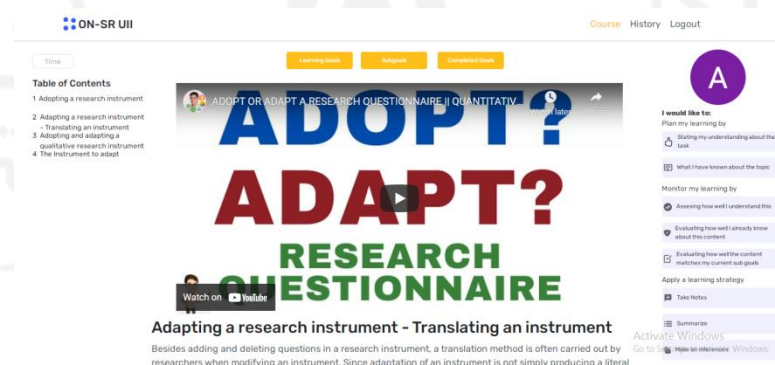
```
public function listCourse($id) {
    if (Auth::check()) {

        request()->session()->put('idTopik', $id);
        $konten = DB::table('konten')->where('topik', $id)->get();
        return view('mahasiswa/listCourse', ['konten'=>$konten]);

    }else{
        return redirect('/logout');
    }
}
```

Gambar 4.21 Kode sumber untuk menampilkan daftar materi

c. Halaman belajar mahasiswa



Gambar 4.22 Halaman belajar mahasiswa

Untuk menampilkan data pada Gambar 4.22 menggunakan fungsi *pilihMateri*. Pada *table of contents* akan menampilkan semua materi sesuai dengan topik yang dipilih menggunakan kode pada Gambar 4.23.

```

$idTopik = request()->session()->get('idTopik');
$konten = DB::table('konten')->where('topik',$idTopik)->get();

```

Gambar 4.23 Kode sumber untuk mengambil materi

Untuk menampilkan data *subgoal* dari *database* menggunakan kode pada Gambar 4.24, *subgoal* yang ditampilkan adalah *subgoal* yang keterangannya masih tanda (-), keterangan tersebut bisa berubah menjadi (komplet) jika mahasiswa mengklik *completed goals*.

```

$idUser = Auth::user()->id;
$subgoal = DB::table('tujuan')->where(['keterangan'=>'-', 'materi'=>$id,
'users'=>$idUser ])->get();

```

Gambar 4.24 Kode sumber untuk menampilkan *subgoals*

Untuk menampilkan data materi dari gambar atau video dan materi penjelasannya menggunakan kode pada Gambar 4.25.

```

$konten1 = DB::table('konten')->where('id',$id)->get();

```

Gambar 4.25 Kode sumber untuk menampilkan materi terpilih

d. Chat

Untuk mengirim *chat* dari mahasiswa menggunakan kode pada Gambar 4.26, jadi pada fungsi *kirimChat* ini akan mengambil dari *id* materi, pesan, *id user* dan kode, kemudian akan disimpan pada *database* di tabel *chat*. Data tersebut nantinya akan ditampilkan pada *chat* dosen dan *chat* mahasiswa. Untuk kode digunakan sebagai pembeda antara *user* mahasiswa dan *user* dosen, mahasiswa kodenya 0 sedangkan dosen kodenya 1.

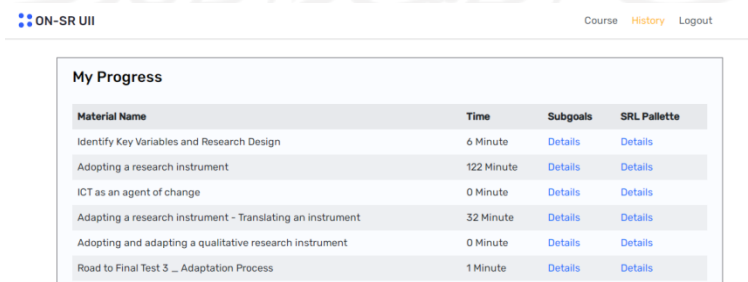
```

$idUser = Auth::user()->id;
$idMateri = $request->session()->get('idMateri');
DB::table('chat')->insert([
    'pesan' => $request->chat,
    'kode' => 0,
    'users' => $idUser,
    'materi' => $idMateri,
    'status' => "belum"]);
return redirect('/pilihMateri/'.session()->get('idMateri'));

```

Gambar 4.26 Kode sumber untuk mengirim *chat*

e. History



Material Name	Time	Subgoals	SRL Palette
Identify Key Variables and Research Design	6 Minute	Details	Details
Adopting a research instrument	122 Minute	Details	Details
ICT as an agent of change	0 Minute	Details	Details
Adapting a research instrument - Translating an instrument	32 Minute	Details	Details
Adopting and adapting a qualitative research instrument	0 Minute	Details	Details
Road to Final Test 3 ... Adaptation Process	1 Minute	Details	Details

Gambar 4.27 Halaman *history* belajar

Pada *history*, seperti pada Gambar 4.27 data yang ditampilkan, yaitu judul materi, total waktu belajar, isian *subgoals* dan isian SRL Palette. Tampilan tersebut dapat digunakan mahasiswa untuk melihat *history* belajarnya dan untuk mengetahui materi yang belum dipelajarinya. Untuk menampilkan *progress*, seperti pada Gambar 4.27 menggunakan kode pada Gambar 4.28, gambar tersebut adalah fungsi *historySiswa*. Menggunakan *join* antara tabel waktu dan tabel *konten*.

```

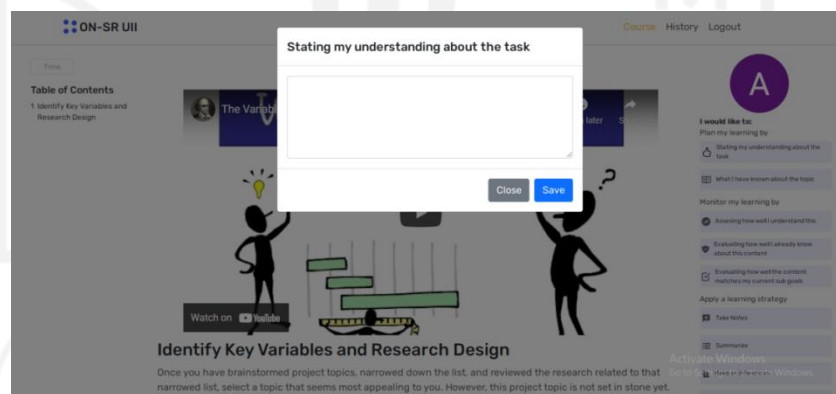
$siswa = Auth::user()->id;
$materi = DB::table('waktu')
    ->join('konten','waktu.materi','=','konten.id')
    ->select('waktu.waktu', 'konten.judul_materi','konten.id')
    ->where('waktu.users','=','',$siswa)
    ->get();
return view('/mahasiswa/riwayatSiswa',['materi'=>$materi]);

```

Gambar 4.28 Kode sumber untuk menampilkan *history*

Penjelasan pada Gambar 4.28, yaitu pada *konten.judul_materi* akan menampilkan semua judul materi yang pernah atau telah dipelajari oleh mahasiswa, pada *waktu.waktu* akan menampilkan waktu belajarnya dan *konten.id* akan mengambil id dari materi. Pada bagian *subgoals* akan menggunakan fungsi *subgoals* untuk menampilkan *subgoal* dari materi tersebut dan pada bagian SRL Palette akan menggunakan fungsi SRL Palette untuk menampilkan isian *plan*, *monitor* dan *strategy*.

f. Mengisi SRL Palette



Gambar 4.29 Tampilan *form* SRL Palette

Menambahkan isian pada *plan stating my understanding about the task* dengan menggunakan kode pada Gambar 4.30, kode ini hampir sama dengan *form* isian untuk *monitor* dan *strategy*. Setelah mahasiswa klik *save* maka fungsi *tambahPemahaman* akan mulai bekerja dengan mengecek apakah mahasiswa pernah mengisi sebelumnya, jika sudah pernah mengisi maka fungsi akan bekerja untuk melakukan *update* data pada *database* perencanaan sesuai dengan materi, jika belum pernah maka akan langsung dimasukkan ke dalam *database* perencanaan.

```

$messages = [
    'required' => 'can not be empty'
];
$this->validate($request, [
    'EvaPemahaman' => 'required'
], $messages);

$idMateri = $request->session()->get('idMateri');
$idUser = Auth::user()->id;
$pemantauan = DB::table('pemantauan')
    ->where(['materi'=>$idMateri, 'users'=>$idUser])->first();
if(empty($pemantauan)){
    DB::table('pemantauan')->insert([
        'evaluasi_pemahaman' => strip_tags($request->EvaPemahaman),
        'evaluasi_pengetahuan' => '',
        'evaluasi_tujuan' => '',
        'materi' => $idMateri,
        'users' => $idUser]);
    return redirect('/pilihMateri/'.session()->get('idMateri'));
}else{
    DB::table('pemantauan')
        ->where(['materi'=>$idMateri, 'users'=>$idUser])
        ->update(['evaluasi_pemahaman' => strip_tags($request->EvaPemahaman)]);
    return redirect('/pilihMateri/'.session()->get('idMateri'));
}

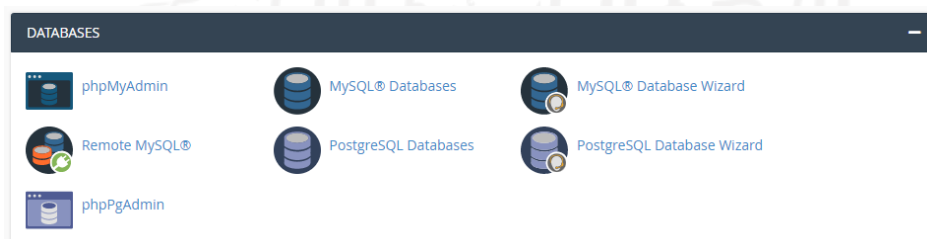
```

Gambar 4.30 Kode sumber untuk menyimpan isian SRL Palette

4.2 Hosting Website

Setelah *website* jadi maka selanjutnya melakukan *hosting* agar *website* bisa diakses oleh dosen atau mahasiswa dari mana pun dan kapan pun. Sesuai dengan kebutuhan di awal mengenai *hosting* maka pada *website* ini memutuskan untuk menggunakan *hosting* dari IDwebhost.

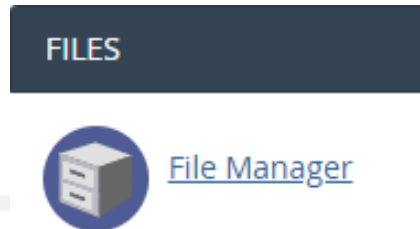
Membuka cPanel dan memilih *MySQL Database* pada Gambar 4.31 dan membuat *database* serta pengaturannya, kemudian melakukan *import* terhadap *database* yang telah dibuat pada *localhost* ke cPanel phpMyAdmin pada Gambar 4.31. Untuk melihat isian *user* pada *database* bisa membuka pada phpMyAdmin.



Gambar 4.31 Database di cPanel

Project laravel akan diunggah ke *file manager*, Gambar 4.32. Sebelum diunggah *folder* laravel akan di zip terlebih dahulu. Ada 2 zip yang akan diunggah, zip pertama adalah zip

yang berisi hanya dari *folder public* laravel, zip kedua adalah zip yang berisi semua *file* dan *folder* laravel, kecuali *folder public*. Untuk zip *public* laravel diletakkan di *public_html*, kemudian lakukan *extract* pada zip, hasilnya akan seperti pada Gambar 4.33.



Gambar 4.32 File Manager di cPanel

Name	Size	Last Modified	Type	Permissions
assets	99 bytes	Dec 22, 2021, 11:20 PM	httpd/unix-directory	0755
materi	224 bytes	Dec 23, 2021, 2:13 AM	httpd/unix-directory	0755
public	6 bytes	Nov 26, 2021, 12:28 PM	httpd/unix-directory	0755
.htaccess	716 bytes	Nov 26, 2021, 11:18 AM	text/x-generic	0644
error_log	624 bytes	Nov 26, 2021, 12:21 PM	text/x-generic	0644
favicon.ico	0 bytes	Sep 30, 2021, 10:23 AM	image/x-generic	0644
index.php	1.71 KB	Nov 26, 2021, 11:15 AM	text/x-generic	0644
public.zip	4.6 MB	Nov 26, 2021, 10:57 AM	package/x-generic	0644
robots.txt	24 bytes	Sep 30, 2021, 10:23 AM	text/plain	0644
web.config	1.16 KB	Sep 30, 2021, 10:23 AM	text/x-generic	0644

Gambar 4.33 Lokasi *folder public* laravel

Kemudian membuat *folder* baru *onsruii* yang diletakkan di */home*. *Folder* tersebut akan digunakan untuk mengunggah zip kedua yang berisi semua *folder* dan *file* laravel kecuali *folder public*, kemudian lakukan *extract*, hasilnya Gambar 4.34. Selanjutnya melakukan penyesuaian pada bagian *.env*. Pertama, *setting database*, yaitu pada *DB_DATABASE*, *DB_USERNAME* and *DB_PASSWORD* sesuaikan dengan *database* yang telah dibuat pada *MySQL Database*. Masih pada *.env* ubah *APP_URL* menjadi alamat web *onsruii* yang tadinya berisi alamat *local host*. Kedua, melakukan *setting* pada *folder public_html* di *file index.php* dibagian */vendor/autoload.php* dan */bootstrap/app.php*, dengan menambahkan nama *folder* laravel (*onsruii*) menjadi */onsruii/vendor/autoload.php* dan */onsruii/bootstrap/app.php*.

Name	Size	Last Modified	Type	Permissions
app	82 bytes	Nov 26, 2021, 11:14 AM	httpd/unix-directory	0755
bootstrap	34 bytes	Nov 26, 2021, 11:14 AM	httpd/unix-directory	0755
config	282 bytes	Nov 26, 2021, 11:14 AM	httpd/unix-directory	0755
database	74 bytes	Nov 26, 2021, 11:14 AM	httpd/unix-directory	0755
resources	64 bytes	Dec 22, 2021, 11:23 PM	httpd/unix-directory	0755
routes	75 bytes	Nov 26, 2021, 11:14 AM	httpd/unix-directory	0755
storage	46 bytes	Nov 26, 2021, 11:14 AM	httpd/unix-directory	0755
tests	83 bytes	Nov 26, 2021, 11:14 AM	httpd/unix-directory	0755
vendor	4 KB	Nov 26, 2021, 11:14 AM	httpd/unix-directory	0755
.editorconfig	258 bytes	Sep 30, 2021, 10:23 AM	text/x-generic	0644
.env	1.24 KB	Nov 26, 2021, 11:17 AM	text/x-generic	0644
.env.example	869 bytes	Sep 30, 2021, 10:23 AM	text/x-generic	0644
.gitattributes	111 bytes	Sep 30, 2021, 10:23 AM	text/x-generic	0644
.gitignore	207 bytes	Sep 30, 2021, 10:23 AM	text/x-generic	0644
.styleci.yml	194 bytes	Sep 30, 2021, 10:23 AM	text/x-generic	0644

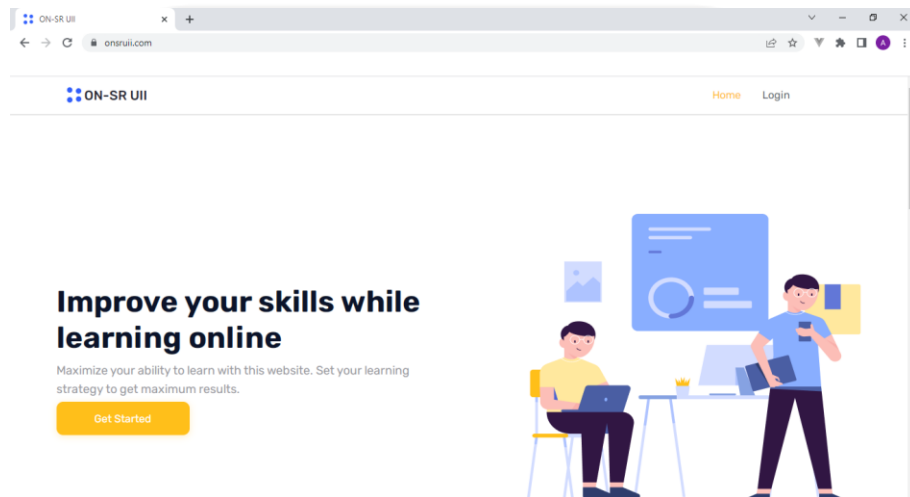
Gambar 4.34 Lokasi *folder onsruii*

4.3 Tampilan *Website* ON-SRUII

Di bawah ini adalah tampilan *website* yang telah dibuat berdasarkan hasil dari *analysis* kebutuhan *user*. Tampilan tersebut sudah bisa diakses dengan internet. Tampilan *website* dibagi menjadi dua, yaitu tampilan dosen dan tampilan mahasiswa.

a. Halaman Dosen

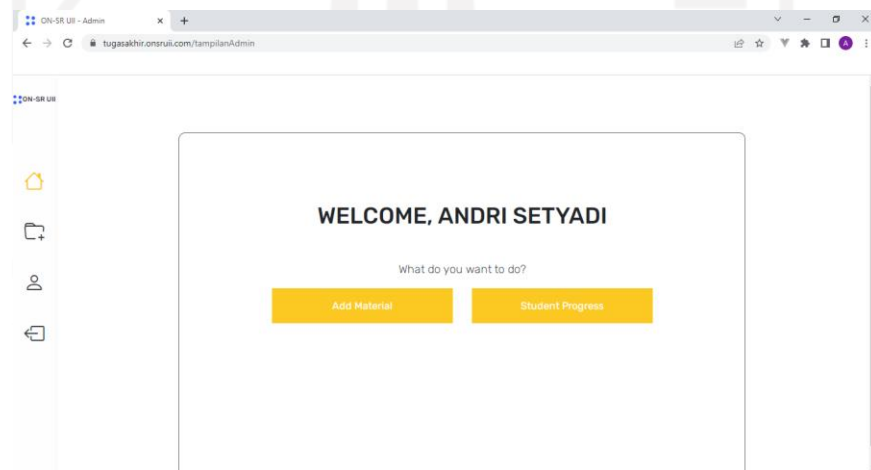
1. Membuka www.onsruii.com



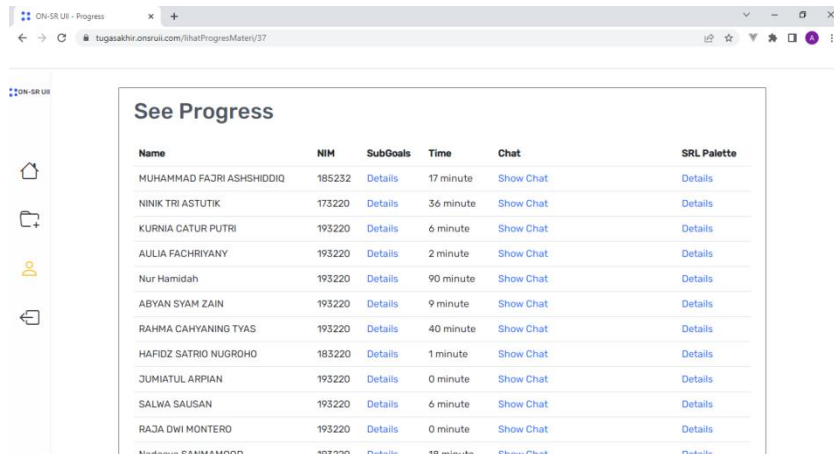
Gambar 4.35 Halaman beranda *website*

2. Masuk ke halaman dosen

Setelah berhasil *login* maka akan muncul halaman *home* dari dosen.



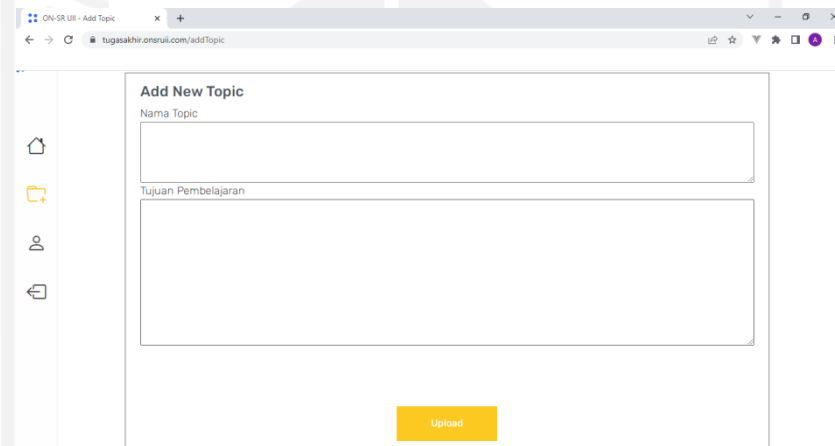
Gambar 4.36 Halaman *home* dosen

4. Mengklik “*student progress*”


Name	NIM	SubGoals	Time	Chat	SRL Palette
MUHAMMAD FAJRI ASHSHIDDIQ	185232	Details	17 minute	Show Chat	Details
NINIK TRI ASTUTIK	173220	Details	36 minute	Show Chat	Details
KURNIA CATUR PUTRI	193220	Details	6 minute	Show Chat	Details
AULIA FACHRIYANY	193220	Details	2 minute	Show Chat	Details
Nur Hamidah	193220	Details	90 minute	Show Chat	Details
ABYAN SYAM ZAIN	193220	Details	9 minute	Show Chat	Details
RAHMA CAHYANING TYAS	193220	Details	40 minute	Show Chat	Details
HAFIDZ SATRIO NUGROHO	183220	Details	1 minute	Show Chat	Details
JUMIATUL ARPIAN	193220	Details	0 minute	Show Chat	Details
SALWA SAUSAN	193220	Details	6 minute	Show Chat	Details
RAJA DWI MONTERO	193220	Details	0 minute	Show Chat	Details
Nadeemva SANMAMOOD	193220	Details	18 minute	Show Chat	Details

Gambar 4.37 Halaman *progress* mahasiswa

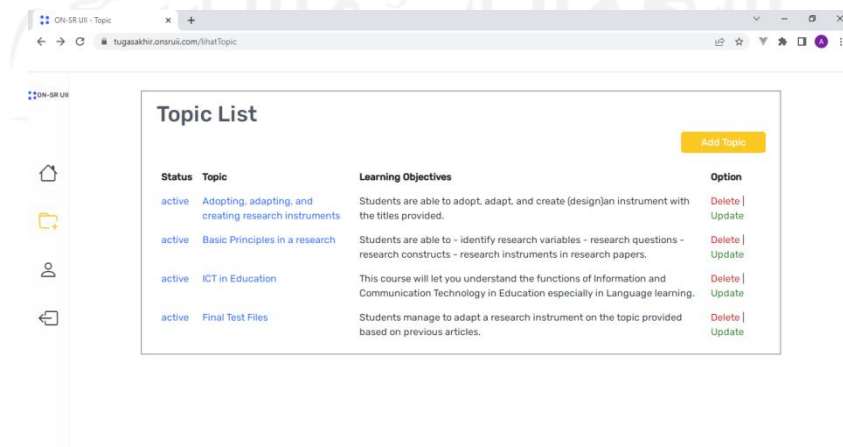
5. Tambah topik



Gambar 4.38 Halaman tambah topik

6. Hapus topik

Untuk menghapus topik langsung klik *delete* di bagian *option*, sesuai dengan urutan topik.



Status	Topic	Learning Objectives	Option
active	Adopting, adapting, and creating research instruments	Students are able to adopt, adapt, and create (design)an instrument with the titles provided.	Delete Update
active	Basic Principles in a research	Students are able to - identify research variables - research questions - research constructs - research instruments in research papers.	Delete Update
active	ICT in Education	This course will let you understand the functions of Information and Communication Technology in Education especially in Language learning.	Delete Update
active	Final Test Files	Students manage to adapt a research instrument on the topic provided based on previous articles.	Delete Update

Gambar 4.39 Halaman daftar topik

7. Update topik

Gambar 4.40 Halaman *update* topik

8. Tambah materi

Gambar 4.41 Halaman tambah materi

9. Hapus materi

Untuk menghapus materi bisa langsung klik *delete* di bagian kanan.

Gambar 4.42 Halaman daftar materi

10. Update materi

Material Update

Nama Materi
The instrument to adapt

Materi

Paragraph

After reading the articles and finding the construct of the variable, now please consider these!

1. Which specific instrument will you use in your research?
2. What are your reasons to choose the instrument?
3. How do you plan to adapt the instrument?

If you need to learn more about instrument adaptation, you can check our resources [here!](#)

Link Video

Tambahkan Gambar

Pilih File Tidak ada file yang dipilih

Upload

Gambar 4.43 Halaman *update materi*

11. Mengirim pesan

RIWAYAT CHAT

Ma saya masih bingung tentang construct. Kalau di artikel, dimana kita bisa melihat konstruk yang dipakai oleh peneliti?

Kirim

Gambar 4.44 Halaman pesan

12. Isian SRL Palette mahasiswa

I would like to plan my learning by

- Stating my understanding about the task
I need to watch the video on research variables.
- What I have known about the topic
Variable is a component in the research that we want to measure.

I would like to monitor my learning by

- Assessing how well I understand this
Cukup paham - variable - construct Need improvement - choosing the instrument
- Evaluating how well I already know about this content
- Evaluating how well the content matches my current sub-goal
Sesuai I Need more resources

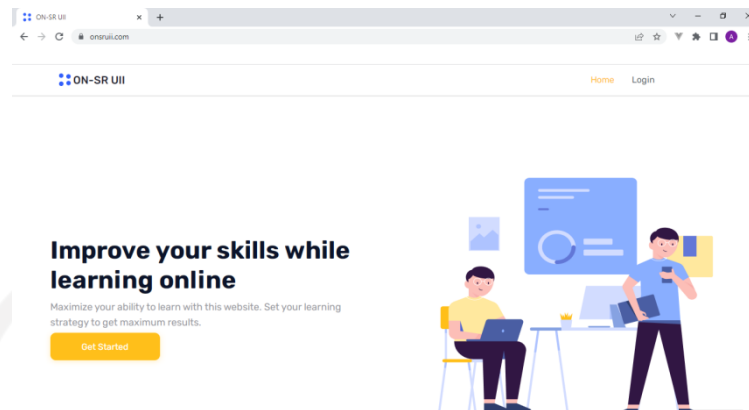
Apply a learning strategy

- Take notes
variable is ... Kinds of variables ...
- Summarize
Variabel terikat yang akan kita buat instrumennya.
- Make an inference
Variabel independen: Variabel yang mempengaruhi Variabel dependen: Variabel yang dipengaruhi. Kadang variabel independent juga perlu diukur.

Gambar 4.45 Halaman isian SRL Palette

b. Halaman Mahasiswa

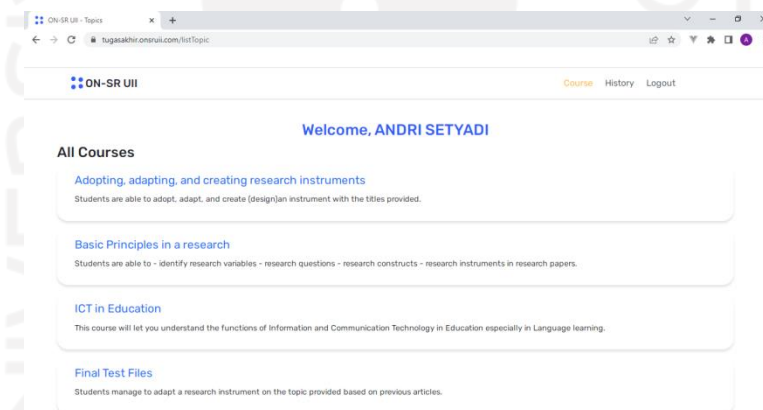
1. Membuka www.onsrui.com



Gambar 4.46 Halaman beranda *website*

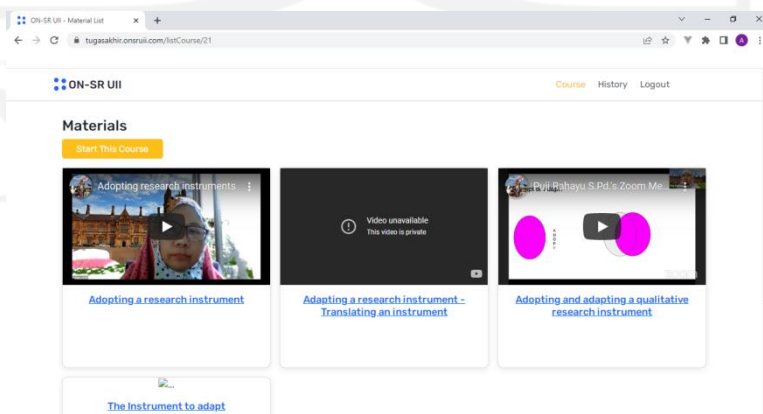
2. Masuk ke halaman mahasiswa

Setelah berhasil *login* maka akan muncul halaman *home* dari mahasiswa.



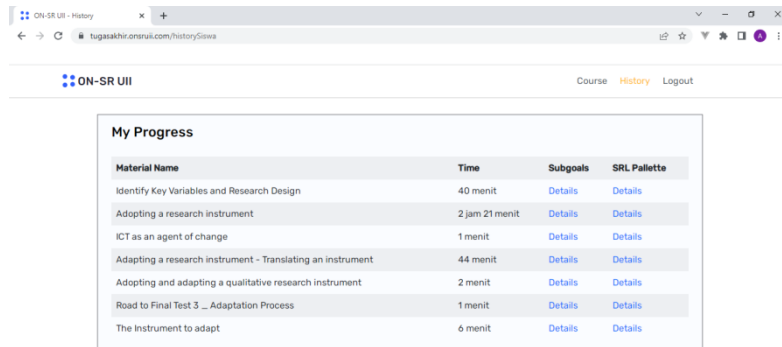
Gambar 4.47 Halaman daftar topik

3. Daftar materi setelah memilih topik



Gambar 4.48 Halaman daftar materi

4. History

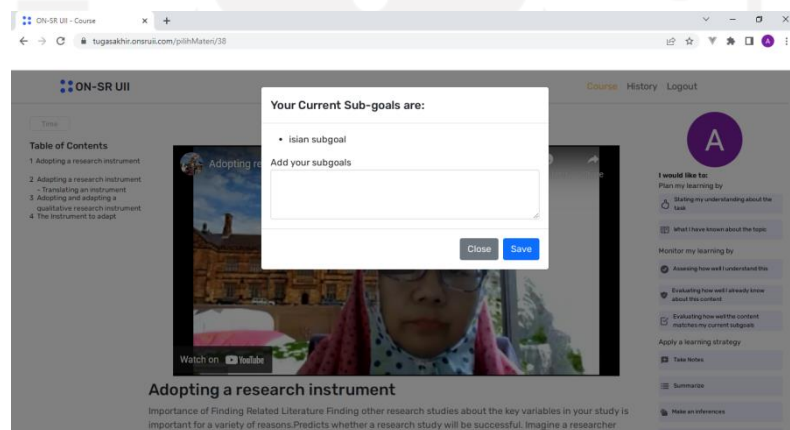


Material Name	Time	Subgoals	SRL Palette
Identify Key Variables and Research Design	40 menit	Details	Details
Adopting a research instrument	2 jam 21 menit	Details	Details
ICT as an agent of change	1 menit	Details	Details
Adapting a research instrument - Translating an instrument	44 menit	Details	Details
Adopting and adapting a qualitative research instrument	2 menit	Details	Details
Road to Final Test 3 _ Adaptation Process	1 menit	Details	Details
The instrument to adapt	6 menit	Details	Details

Gambar 4.49 Halaman *history* belajar

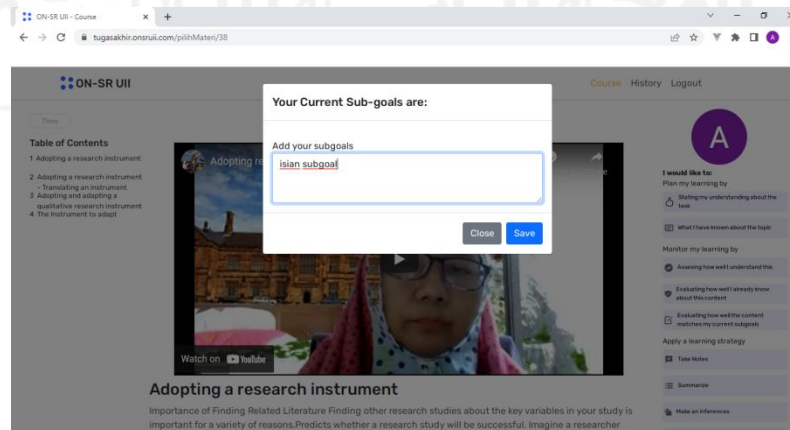
5. Complete subgoals

Jika *button complete subgoal* diklik maka *list subgoal* tersebut akan hilang di halaman ini (*subgoal* selesai) dan akan dimunculkan pada halaman *history*.



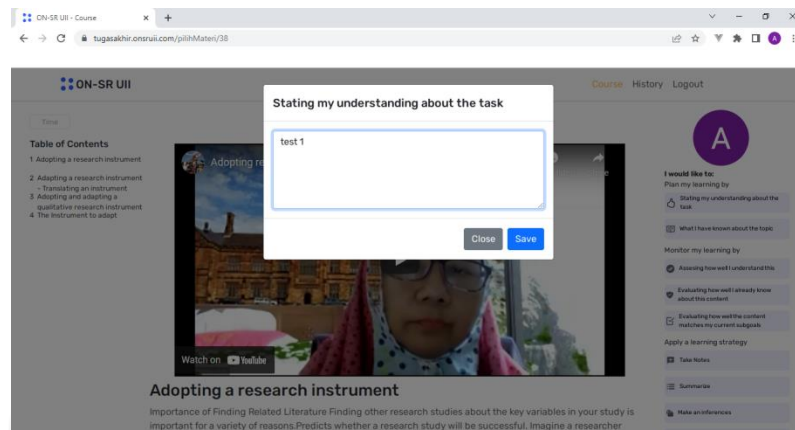
Gambar 4.50 Tampilan daftar *subgoal*

6. Menambahkan *subgoals*



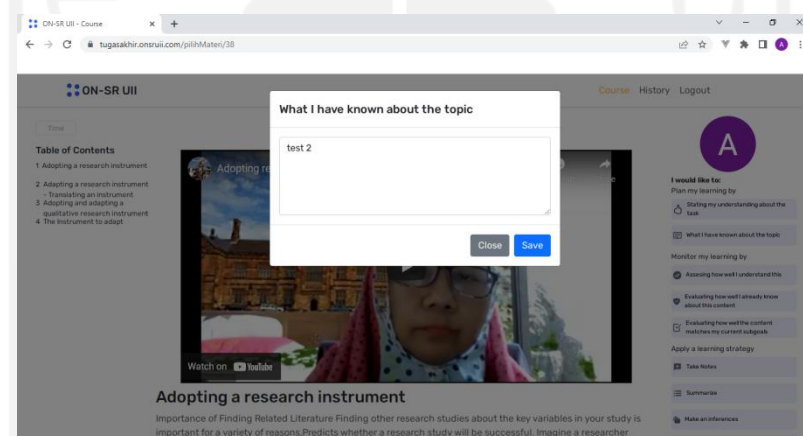
Gambar 4.51 Tampilan *form subgoals*

7. Form add Stating my understanding about the task



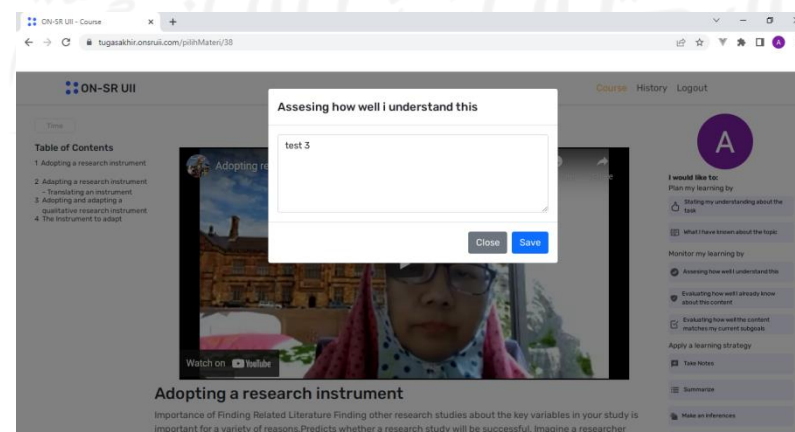
Gambar 4.52 Tampilan form *Stating my understanding about the task*

8. Form add What I have known about the topic



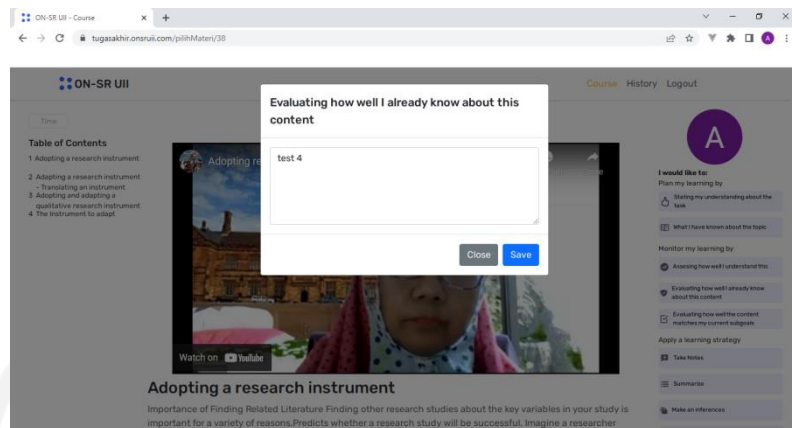
Gambar 4.53 Tampilan form *What I have known about the topic*

9. Form add Assesing how well I understand this



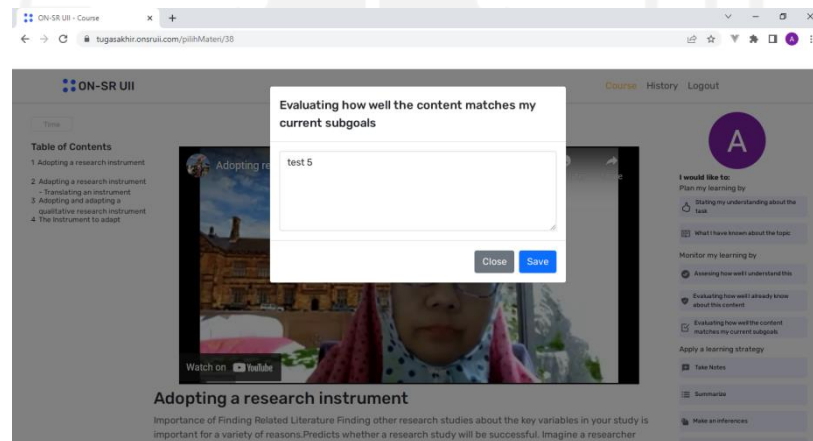
Gambar 4.54 Tampilan form *assesing how well I understand this*

10. Form add Evaluating how well I already know about this content



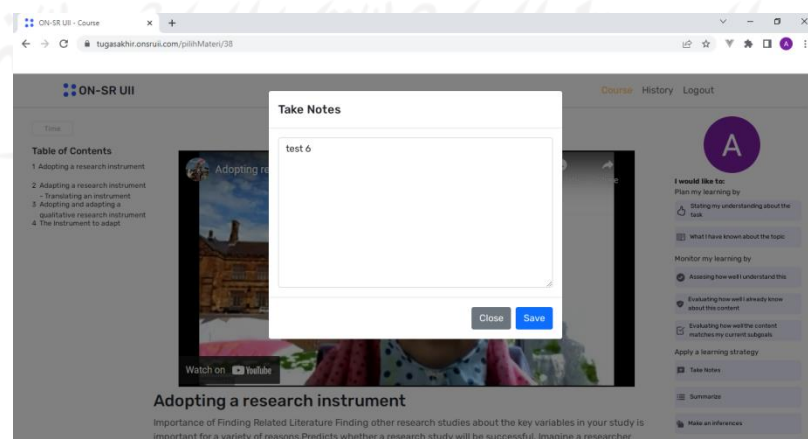
Gambar 4.55 Tampilan form evaluating how well I already know about this content

11. Form add Evaluating how well the content matches my current subgoals



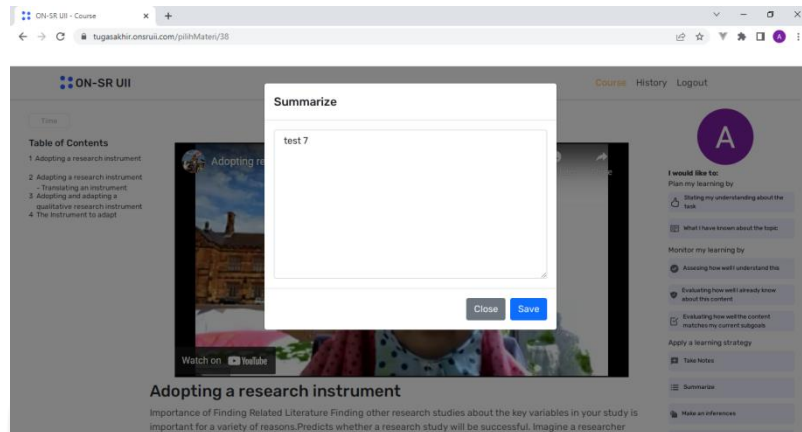
Gambar 4.56 Tampilan form evaluating how well the content matches my current subgoals

12. Form add Take Notes



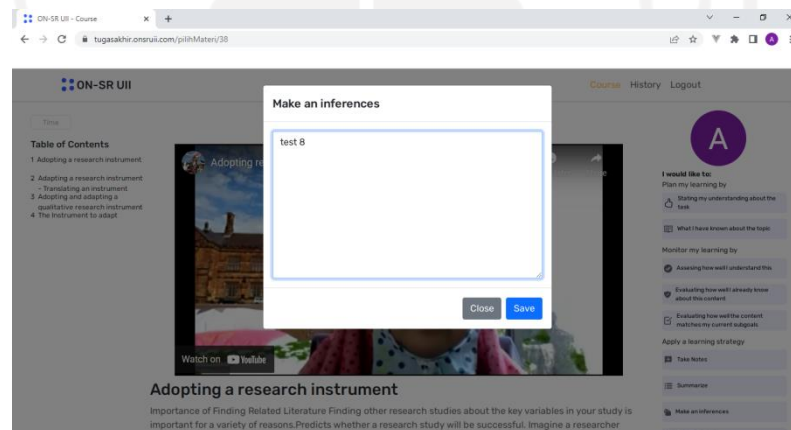
Gambar 4.57 Tampilan form take notes

13. Form add Summarize



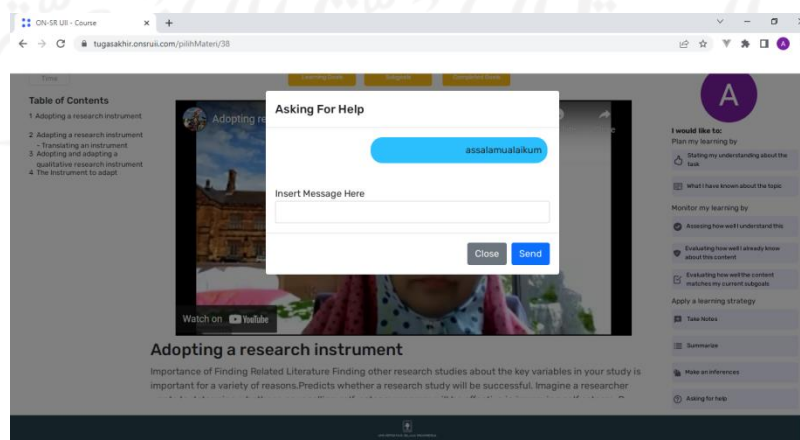
Gambar 4.58 Tampilan form summarize

14. Form add Make an inferences



Gambar 4.59 Tampilan form make an inferences

15. Form add Chat



Gambar 4.60 Tampilan form chat

4.4 Pengujian *Website ON-SRUII*

Pengujian *website* akan menguji pada bagian fungsionalitas sistem. Menggunakan *black box testing*. Daftar pengujian sesuai dengan desain pada bab sebelumnya. Setelah dosen dan mahasiswa mencoba web kemudian akan mengisi pertanyaan lewat *google form*.

Pengujian pertama peneliti melakukan sendiri dengan cara memeriksa semua bagian *website* apakah ada terjadi *error* atau tidak. Ternyata masih ditemukan *error* pada bagian dosen, ketika dosen tidak mengisi *link* maupun gambar seharusnya tidak terjadi *error*, tetapi malah *error*, maka dilakukan perbaikan pada kode tersebut. Setelah diperbaiki dan memeriksa lagi web sudah tidak terjadi *error*. Maka selanjutnya mengujikan *website* pada dosen dan mahasiswa.

Berikut ini adalah hasil pengujian pada web ON-SRUII:

Tabel 4.1 Hasil pengujian fungsionalitas dari *user* dosen

No	Skenario	Jumlah	
		Berhasil	Tidak
1	Membuka www.onsruii.com	3	0
2	Masuk ke halaman dosen	3	0
3	Mengklik " <i>student progress</i> "	3	0
4	Menambahkan Topik	3	0
5	Menghapus Topik	3	0
6	Memperbarui Topik	3	0
7	Menambahkan Materi	3	0
8	Menghapus Materi	3	0
9	Memperbarui Materi	3	0
10	Mengirim Pesan	3	0
11	Mengklik " <i>details SRL Palette</i> "	3	0

Pada pengujian web bagian tampilan dosen dilakukan oleh satu dosen dan 2 mahasiswa. Penguji mencoba web dengan menjalankan berbagai fitur dan mengisi *form* yang ada di *website*. Saat pengujian dilakukan tidak ditemukan *error* atau kesalahan pada *website* yang ditunjukkan dari hasil *form* yang diisi setelah pengujian dilakukan. Penguji dosen berasal dari jurusan Pendidikan Bahasa Inggris, Fakultas Psikologi dan Ilmu Sosial Budaya,

Universitas Islam Indonesia, sedangkan mahasiswa berasal dari jurusan Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia. Dosen dan mahasiswa berhasil mengisi dan menjalankan pada semua skenario yang ada di tabel. Dengan ini maka fungsionalitas dari web ON-SRUII bagian dosen berfungsi sebagaimana mestinya dan tidak terjadi *error* saat pemrosesannya.

Tabel 4.2 Hasil pengujian fungsionalitas dari *user* mahasiswa

No	Skenario	Jumlah	
		Berhasil	Tidak
1	Membuka www.onsruii.com	32	0
2	Masuk ke halaman mahasiswa	32	0
3	Memilih topik	32	0
4	Mengklik " <i>history</i> "	32	0
5	Mengklik " <i>complete</i> "	32	0
6	<i>Add subgoal</i>	32	0
7	<i>Add Stating my understanding about the task.</i>	32	0
8	<i>Add What I have known about the topic</i>	32	0
9	<i>Add Assesing how well I understand this</i>	32	0
10	<i>Add Evaluating how well I already know about this content</i>	32	0
11	<i>Add Evaluating how well the content matches my current subgoals</i>	32	0
12	<i>Add Take Notes</i>	32	0
13	<i>Add Summarize</i>	32	0
14	<i>Add Make an inferences</i>	32	0
15	<i>Add Chat</i>	32	0

Mahasiswa yang mencoba menjalankan web ON-SRUII dan mengisi *form* berjumlah 32 orang dari angkatan 2019 jurusan Pendidikan Bahasa Inggris, Fakultas Psikologi dan Ilmu Sosial Budaya, Universitas Islam Indonesia. Pada bagian ini mahasiswa akan mencoba web dari sisi mahasiswa. Setelah mahasiswa mencoba web dan melakukan pengisian data di dalam *website* maka selanjutnya mahasiswa akan mengisi *form* pertanyaan, berdasarkan isian *form* dari 32 mahasiswa, menyatakan bahwa fungsionalitas sistem dapat berjalan dengan baik sesuai dengan skenario yang dibuat, tidak ada kesalahan dan tidak ada *error* yang terjadi.

BAB V KESIMPULAN DAN SARAN

4.5 Kesimpulan

Berikut adalah kesimpulan berdasarkan penelitian yang telah dilakukan pada pengembangan aplikasi web ON-SRUII dari tahap awal sampai tahap pengujian kepada *user*:

- a. Penggunaan *framework* laravel untuk membangun *website* berhasil, *website* dapat berjalan sesuai dengan kebutuhan *user*. Pada bagian tampilan yang akan ditampilkan ke *user* diletakkan pada *view*, untuk fungsi-fungsi diletakkan di *controllers* dan untuk penghubung antar halaman diletakkan di *web.php*.
- b. *Website* berhasil di *hosting* sehingga dosen dan mahasiswa bisa membuka web dari manapun dengan menggunakan internet, *website* telah digunakan dosen untuk mengajar mata kuliah RIDA.
- c. Pengguna dari kalangan mahasiswa telah berhasil untuk menggunakan fitur *subgoal*, *chat* dan SRL Palette. Menggunakan SRL Palette untuk mengisi *plan*, *monitor* dan *strategy*. *Plan* terdapat dua isian, yaitu *Stating my understanding about the task* dan *What I have known about the topic*. *Monitor* terdapat tiga isian, yaitu *Assesing how well I understand this*, *Evaluating how well I already know about this content* dan *Evaluating how well I already know about this content*. *Strategy* terdapat empat isian, yaitu *Take notes*, *Summarize*, *Make an inferences* dan *Asking for help*. Isian SRL Palette digunakan untuk mengetahui sejauh mana pengetahuan atau pemahaman mahasiswa terhadap materi yang sedang dipelajarinya. Menggunakan *subgoal* untuk mengisi tujuan belajar dari mahasiswa di setiap materinya. Menggunakan *chat* untuk mengisi pesan yang akan dikirimkan ke dosen.
- d. Pengguna dari kalangan dosen telah berhasil untuk memberikan informasi terkait topik dan materi yang harus dipelajari.
- e. Pengguna dari kalangan dosen telah berhasil mendapatkan informasi mengenai *progress* belajar dari para mahasiswa mulai dari waktu belajar, isian tujuan belajar dan isian SRL Palette. Hasil *progress* belajar mahasiswa tersebut dapat digunakan dosen untuk melakukan evaluasi.

4.6 Saran

Penulis sadar bahwa pengembangan aplikasi web ON-SRUII masih terdapat kekurangan. Semoga untuk kedepannya aplikasi web ON-SRUII dapat dikembangkan dengan lebih baik lagi. Saran jika akan dilakukan pengembangan, yaitu:

- a. Membuat fitur untuk dosen dapat memberikan kuis atau ujian di web.
- b. Mahasiswa dapat berkomunikasi dengan mahasiswa lainnya secara bersamaan pada web untuk membahas materi yang belum paham, seperti *group chat*.



DAFTAR PUSTAKA

- Adhi, I. K., Putra, D., Arini, N. W., & Sudarma, I. K. (2019). *Pengaruh Model Self Regulated Learning terhadap Hasil Belajar IPA Siswa*. 3(3), 258–266. <https://ejournal.undiksha.ac.id/index.php/IJEE/article/view/19406>.
- Aisyah, S. (2017). *Jurnal Pendidikan Ilmu-Ilmu Sosial Strategy Self Regulated Learning pada Mahasiswa Fakultas Psikologi Universitas Medan Area*. 9(2), 202–212. <https://jurnal.unimed.ac.id/2012/index.php/jupiis/article/view/8246>.
- Aminah, M., & Maulida, I. (2020). Pengaruh Self-Regulated Learning Terhadap Keterampilan Berbicara Bahasa Inggris Siswa. *Biormatika* : 6(1), 132–138. <http://ejournal.unsub.ac.id/index.php/FKIP/article/view/702>.
- Ana, A., & Achdiani, Y. (2017). Penerapan Self Regulated Learning Berbasis Internet Untuk Meningkatkan Kemandirian Belajar Mahasiswa. *Innovation of Vocational Technology Education*, 11(1), 15–22. <https://doi.org/10.17509/invotec.v11i1.4835>.
- Endra, R. Y., Aprilinda, Y., Dharmawan, Y. Y., & Ramadhan, W. (2021). Analisis Perbandingan Bahasa Pemrograman PHP Laravel dengan PHP Native pada Pengembangan Website. *EXPERT: Jurnal Manajemen Sistem Informasi Dan Teknologi*, 11(1), 48. <https://doi.org/10.36448/expert.v11i1.2012>.
- Haryana, K. (2019). Penerapan Agile Development Methods Dengan Framework Scrum Pada Perancangan Perangkat Lunak Kehadiran Rapat Umum Berbasis Qr-Code. *Jurnal Computech & Bisnis*, 13(2), 70–79. <http://doi.org/10.5281/zenodo.3631045>
- Hidayat, H., Hartono, & Sukiman. (2017). Pengembangan Learning Management System (LMS) Untuk Bahasa Pemrograman PHP. *Jurnal Ilmiah Core IT: Community Research Information Technology*, 5(1), 20–29. <http://ijcoreit.org/index.php/coreit/article/view/11>.
- Lutfiana, R. D., & Wibawa, S. C. (2019). Pengembangan Media Pembelajaran E-learning pada Mata Pelajaran Komputer dan Jaringan Dasar dengan Framework Laravel di SMK. In *Journal of Chemical Information and Modeling* (Vol. 4, Issue 1, pp. 90–95). <https://ejournal.unesa.ac.id/index.php/it-edu/article/view/28870>.
- Ma'ruf, A., Widiartin, T., & Prasetya, N. I. (2017). Sistem Pembelajaran Berbasis Web (E-Learning) Ma Darussalam Jombang. *Melek IT*, 3(2), 49–58. http://melekit.if.uwks.ac.id/index.php/printmelekit/article/view/281/pdf_32.
- Mukhid, A. (2008). STRATEGI SELF-REGULATED LEARNING (Perspektif Teoritik). *Journal of Educational Psychology*, 82(1), 33–40.

<http://ejournal.iainmadura.ac.id/index.php/tadris/article/view/239>.

- Prihandani, K. (2016). Tinjauan Kualitas Pengembangan Sistem Informasi Dengan Metode Agile . *Tinjauan Kualitas Pengembangan Sistem Informasi Dengan Metode Agile .*, October, 1–6. https://www.researchgate.net/profile/Kamal-Prihandani/publication/309318094_Tugas_review_jurnal/links/5809883d08ae993dc050a4b0/Tugas-review-jurnal.pdf
- Rabbani, I., & Krisnanik, E. (2020). E- Commerce Perlengkapan Haji Dan Umroh Berbasis Web Menggunakan Metode Agile Software Development. *Seminar Nasional Mahasiswa Ilmu Komputer Dan Aplikasinya (SENAMIKA)*, 1(2), 432–443. <https://conference.upnvj.ac.id/index.php/senamika/article/view/584/453>
- Sofwan, A. (2011). Belajar Mysql dengan Phpmyadmin Pendahuluan. *Modul Kuliah Graphical User Interface I (GUI) Di Perguruan Tinggi Raharja*, 1–29. <https://ilmukomputer.org/wp-content/uploads/2011/08/sofwan-ysqldanphpmyadmin.pdf>.
- Somya, R., & Nathanael, T. M. E. (2019). Pengembangan Sistem Informasi Pelatihan Berbasis Web Menggunakan Teknologi Web Service Dan Framework Laravel. *Jurnal Techno Nusa Mandiri*, 16(1), 51–58. <https://doi.org/10.33480/techno.v16i1.164>.
- Susanto, W. E., & Ayu, Y. G. A. (2017). Perancangan E-Learning Berbasis Web Pada SMP Negeri 3 Patuk Gunungkidul Yogyakarta. *Bianglala Informatika*, 5(2). <https://doi.org/10.31294/bi.v5i2.2806>.
- Vianna, M., & Vianna, Y. (2018). Design Thinking and Scrum in the context of digital transformation. *MJV Technology and Innovation Report*. <https://cdn2.hubspot.net/hubfs/2897528/Materials/Ebooks/files/Design%20Thinking%20and%20Scrum%20in%20the%20context%20of%20Digital%20Transformation.pdf>.
- Yauma, A., Fitri, I., & Ningsih, S. (2020). Learning Management System (LMS) pada E-Learning Menggunakan Metode Agile dan Waterfall berbasis Website. *Jurnal JTik (Jurnal Teknologi Informasi Dan Komunikasi)*, 5(3), 323. <https://doi.org/10.35870/jtik.v5i3.190>.
- Yuliano, T. (2017). Pengenalan PHP. *Ilmiu Komputer*, 1–9. <https://ilmukomputer.org/wp-content/uploads/2009/03/triswan-pengenalanphp.pdf>.