

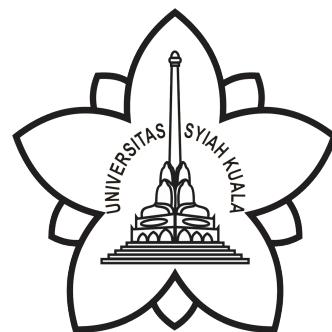
**PENGEMBANGAN FITUR FORUM DISKUSI PADA  
SISTEM APLIKASI PEMBELAJARAN  
STATISTIKA (*RWIKISTAT*) BERBASIS WEBSITE**

**TUGAS AKHIR**

Diajukan untuk melengkapi tugas-tugas dan  
memenuhi syarat-syarat guna memperoleh gelar Sarjana Komputer

Oleh:

**AFFAN IAN AMARA**  
**1908107010038**



**JURUSAN INFORMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS SYIAH KUALA  
DARUSSALAM, BANDA ACEH  
APRIL, 2024**

## **PENGESAHAN SIDANG SARJANA**

### **PENGEMBANGAN FITUR FORUM DISKUSI PADA SISTEM APLIKASI PEMBELAJARAN STATISTIKA (*RWIKISTAT*) BERBASIS WEBSITE**

***DEVELOPMENT OF DISCUSSION FORUM FEATURE IN  
STATISTICS LEARNING APPLICATION SYSTEM (*RWIKISTAT*)  
BASED ON WEBSITE***

Oleh:

Nama : Affan Ian Amara  
NPM : 1908107010038  
Jurusan : Informatika

Menyetujui:

Pembimbing I

Pembimbing II

Kurnia Saputra, S.T., M.Sc.  
NIP. 198003262014041001

Prof. Dr. Hizir Sofyan  
NIP. 196805311993031003

Mengetahui:

Dekan Fakultas MIPA  
Universitas Syiah Kuala,

Koordinator Program Studi,

Prof. Dr. Taufik Fuadi Abidin, S.Si., M.Tech  
NIP. 197010081994031002

Alim Misbullah S.Si., M.S.  
NIP. 198806032019031011

Lulus Sidang Sarjana pada hari Selasa, 2 April 2024

## PERNYATAAN BEBAS PLAGIASI

Saya yang bertanda tangan dibawah ini,

Nama Lengkap : Affan Ian Amara  
Tempat/Tanggal Lahir : Asir-Asir Atas, 9 April 2001  
NPM : 1908107010038  
Program Studi : S1 Informatika  
Fakultas : Matematika dan Ilmu Pengetahuan Alam  
Judul Tugas Akhir : Pengembangan Fitur Forum Diskusi Pada Sistem Aplikasi Pembelajaran Statistika (Rwikistat) Berbasis Website

Menyatakan dengan sesungguhnya bahwa Laporan Tugas Akhir saya dengan judul seperti di atas adalah **Hasil Karya Saya Sendiri** bersama dosen pembimbing dan **Bebas Plagiasi**.

Jika ternyata dikemudian hari terbukti bahwa Laporan Tugas Akhir saya merupakan hasil plagiasi, saya bersedia menerima sanksi yang berlaku di Universitas Syiah Kuala.

Darussalam, 2 April 2024  
Yang Menyatakan,

Affan Ian Amara  
1908107010038

## **SURAT PERNYATAAN**

Yang bertanda tangan dibawah ini,

1. Nama : Affan Ian Amara  
NPM : 1908107010038  
Jurusan/Prodi : Informatika  
Status : Mahasiswa
2. Nama : Kurnia Saputra, S.T., M.Sc.  
NIP : 198003262014041001  
Jurusan/Prodi : Informatika  
Status : Pembimbing I
3. Nama : Prof. Dr. Hizir Sofyan  
NIP : 196805311993031003  
Jurusan/Prodi : Statistika  
Status : Pembimbing II

Dengan ini menyatakan hasil penelitian Tugas Akhir yang berjudul “Pengembangan Fitur Forum Diskusi Pada Sistem Aplikasi Pembelajaran Statistika (Rwikistat)” tidak dipublikasikan hingga batas waktu 5 (lima) tahun.

Demikian surat pernyataan ini dibuat dengan sebenarnya untuk dapat dipergunakan seperlunya.

Darussalam, 2 April 2024  
Yang membuat pernyataan,

Pembimbing I,

Pembimbing II,

Mahasiswa,

Kurnia Saputra, S.T., M.Sc.  
NIP. 198003262014041001

Prof. Dr. Hizir Sofyan  
NIP. 196805311993031003

Affan Ian Amara  
NPM. 1908107010038

Mengetahui:

Koordinator Program Studi  
Universitas Syiah Kuala,

Koordinator TA,

Alim Misbullah S.Si., M.S.  
NIP. 198806032019031011

Alim Misbullah S.Si., M.S.  
NIP. 198806032019031011

## ABSTRAK

Aplikasi Rwikistat adalah sebuah aplikasi pembelajaran yang menyediakan wadah untuk belajar statistika dan *data science* menggunakan bahasa pemrograman R. Pada versi sebelumnya, aplikasi Rwikistat hanya sekedar untuk menjalankan *code* R dan membaca modul. oleh karena itu, dikembangkan kembali aplikasi Rwikistat dengan fitur forum diskusi. Penelitian ini bertujuan untuk mengembangkan fitur forum diskusi dan *Representational State Transfer Application Programming Interface* atau yang biasa disebut REST API pada sistem aplikasi pembelajaran statistika (Rwikistat) berbasis website. Latar belakang penelitian menunjukkan kebutuhan akan platform diskusi langsung di aplikasi, sementara pengembangan REST API bertujuan memperluas aksesibilitas ke aplikasi *mobile*. Metode pengembangan yang digunakan adalah metode SCRUM. Rumusan masalah penelitian ini mencakup pengembangan forum diskusi yang memenuhi kebutuhan pengguna, pembangunan REST API untuk *website* dan aplikasi *mobile*, serta pengujian aplikasi menggunakan *black box testing* dan metode *User Metrics for User Experience* (UMUX). Penelitian ini bertujuan untuk mencakup pengembangan fitur forum diskusi yang memenuhi kebutuhan pengguna, pembangunan REST API, dan pengujian fungsionalitas serta usabilitas aplikasi. Hasil penelitian menunjukkan bahwa forum diskusi yang dikembangkan telah berhasil memenuhi kebutuhan pengguna dengan fitur seperti pertanyaan, komentar, *search*, *like*, dan *bookmark*. REST API, dibangun dengan menggunakan ExpressJS, dapat diakses oleh *website* dan aplikasi *mobile* Rwikistat. Pengujian aplikasi menghasilkan hasil positif, menunjukkan kinerja yang baik dan kepuasan pengguna. Penelitian ini menyimpulkan bahwa pengembangan fitur forum diskusi dan REST API pada Rwikistat memberikan kontribusi positif terhadap fungsionalitas dan usability aplikasi, membuka peluang akses melalui berbagai platform.

**Kata kunci :** Forum diskusi, Aplikasi Rwikistat, Statistika, REST API, *Website, Mobile, Black Box Testing, User Metrics for User Experience*.

## ABSTRACT

*The Rwikistat application is an educational platform that provides a space for learning statistics and data science using the R programming language. In its previous version, Rwikistat merely allowed the execution of R code and module reading. Therefore, the application was redeveloped to include a discussion forum feature. This research aims to develop a discussion forum feature and Representational State Transfer Application Programming Interface or REST API for the statistics learning application (Rwikistat) based on a website. The background of the study indicates the need for a direct discussion platform within the application, while the development of the REST API aims to extend accessibility to mobile applications. The development method employed is the SCRUM methodology. The research problem formulation includes the development of a discussion forum that meets user needs, the construction of a REST API for the website and mobile applications, and application testing using black-box testing and the User Metrics for User Experience (UMUX) method. This study aims to encompass the development of the discussion forum feature to meet user needs, the construction of the REST API, and testing of application functionality and usability. The research findings show that the developed discussion forum successfully meets user needs with features such as questions, comments, search, likes, and bookmarks. The REST API, built using ExpressJS, can be accessed by the Rwikistat website and mobile applications. Application testing yielded positive results, indicating good performance and user satisfaction. In conclusion, this research asserts that the development of the discussion forum feature and REST API in Rwikistat contributes positively to the functionality and usability of the application, providing opportunities for access across various platforms.*

**Keywords :** Discussion Forum, Rwikistat Application, statistics, REST API, Website, Mobile, Black Box Testing, User Metrics for User Experience.

## KATA PENGANTAR

Segala Puji dan Syukur dipanjatkan ke hadirat Allah SWT yang telah memberikan rahmat serta hidayah-Nya, sehingga penulis dapat menyelesaikan Tugas Akhir ini. Selawat dan salam kepada baginda besar Nabi Muhammad SAW yang telah membawa umatnya dari alam jahiliah gelap gulita ke alam islamiah terang benderang dan penuh dengan ilmu pengetahuan.

Tugas Akhir yang berjudul “**Pengembangan Fitur Forum Diskusi Pada Sistem Aplikasi Pembelajaran Statistika (*RWIKISTAT*) Berbasis Website**” ini disusun untuk memenuhi salah satu persyaratan untuk memperoleh gelar sarjana pada program Strata-1 Jurusan Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Syiah Kuala. Tugas Akhir ini dapat diselesaikan dengan lancar atas bantuan dan dukungan dari berbagai pihak. Oleh Sebab itu, penulis hendak mengucapkan rasa terima kasih yang luarbiasa kepada:

1. Ayahanda Qazwin Sofyan dan ibunda Aida Fitri A. yang selalu memberikan kasih sayang dan mendoakan setiap langkah dalam hal apapun. Terima kasih telah memberikan dukungan moril serta materil selama jenjang perkuliahan hingga pelaksanaan penelitian dan juga telah sabar menunggu anak sulungnya menjadi seorang sarjana.
2. Alim Misbullah S.Si., M.S., selaku Koordinator Program Studi Informatika.
3. Bapak Kurnia Saputra, S.T., M.Sc., selaku pembimbing I dan Bapak Prof.Dr. Hizir Sofyan, selaku pembimbing II yang telah sabar membimbing dan selalu bersemanagat dalam memberikan banyak masukan dan arahan kepada penulis dalam penyelesaian Tugas Akhir ini.
4. Bapak Dr. Muhammad Subianto S.Si, M.Si, selaku dosen wali yang selalu terbuka untuk berkonsultasi tentang segala hal mengenai perkuliahan saya.
5. Seluruh Dosen Informatika yang telah memberikan ilmu pengetahuan kepada penulis selama menjalani program studi.
6. Adik penulis, Naurah Rayyani dan Muhammad Danish yang senantiasa mendukung dan selalu menunggu kepulangan abangnya.
7. Yunza Jauhari yang selalu menemani dan memberikan semangat selama menyelesaikan Tugas Akhir, serta selalu meluangkan waktu dan pikiran untuk mendengarkan keluh kesah penulis selama menempuh pendidikan S1 Informatika.

8. Muhammad Nurifai selaku rekan seperjuangan dalam menyelesaikan Tugas Akhir.
9. Teman-teman Informatika 2019.
10. Dan semua pihak yang tidak bisa disebutkan namanya satu per satu yang telah memberikan saran dan semangat dalam menyelesaikan Tugas Akhir ini
11. *Last but not least. I wanna thank me. I wanna thank me for believing in me. I wanna thank me for doing all this hard work. I wanna thank me for having no days off. I wanna thank me for never quitting. I wanna thank me for always being the best. Be ready, this is just the beginning.*

Penulis menyadari segala kekurangan yang terdapat dalam Tugas Akhir ini baik dari segi penulisan maupun cara penyampaian materi yang disajikan. Oleh karena itu, penulis mengharapkan kritikan dan saran yang bersifat membangun dari pembaca yang bertujuan agar penulis dapat memperbaiki kekurangan tersebut ke depannya. Penulis berharap agar Tugas Akhir ini bermanfaat bagi banyak orang di kemudian hari dan dapat berguna bagi perkembangan ilmu pengetahuan dan teknologi.

Banda Aceh, April 2024  
Penulis,

**Affan Ian Amara**

## DAFTAR ISI

<b>PENGESAHAN SIDANG SARJANA</b>	<b>ii</b>
<b>PERNYATAAN BEBAS PLAGIASI</b>	<b>iii</b>
<b>SURAT PERNYATAAN</b>	<b>iv</b>
<b>ABSTRAK</b>	<b>v</b>
<b>ABSTRACT</b>	<b>vi</b>
<b>KATA PENGANTAR</b>	<b>vii</b>
<b>DAFTAR ISI</b>	<b>x</b>
<b>DAFTAR TABEL</b>	<b>xi</b>
<b>DAFTAR GAMBAR</b>	<b>xiii</b>
<b>DAFTAR LAMPIRAN</b>	<b>xiii</b>
<b>BAB I PENDAHULUAN</b>	<b>1</b>
1.1. LATAR BELAKANG . . . . .	1
1.2. RUMUSAN MASALAH . . . . .	2
1.3. TUJUAN PENELITIAN . . . . .	2
1.4. MANFAAT PENELITIAN . . . . .	3
<b>BAB II TINJAUAN KEPUSTAKAAN</b>	<b>4</b>
2.1. FORUM DISKUSI <i>ONLINE</i> . . . . .	4
2.2. <i>WEBSITE</i> . . . . .	5
2.2.1. Framework . . . . .	5
2.3. REACT.JS . . . . .	6
2.4. TYPESCRIPT . . . . .	7
2.5. OPENAI . . . . .	7
2.6. PENETRATION TESTING . . . . .	8
2.7. SCRUM . . . . .	8
2.8. <i>BLACKBOX TESTING</i> . . . . .	10
2.9. <i>USABILITY TESTING</i> . . . . .	10
2.10. <i>USABILITY METRICS FOR USER EXPERIENCE (UMUX)</i> . .	11
2.11. PENELITIAN TERKAIT . . . . .	11
<b>BAB III METODOLOGI PENELITIAN</b>	<b>13</b>
3.1. WAKTU DAN LOKASI PENELITIAN . . . . .	13
3.2. ALAT DAN BAHAN . . . . .	13
3.2.1. Perangkat keras ( <i>Hardware</i> ) . . . . .	13
3.2.2. Perangkat lunak ( <i>Software</i> ) . . . . .	13
3.3. METODE PENELITIAN . . . . .	14

3.3.1. Identifikasi Masalah . . . . .	14
3.3.2. Studi Literatur . . . . .	15
3.3.3. Analisis Kebutuhan . . . . .	15
3.3.4. Perancangan Sistem . . . . .	16
3.3.5. Pengembangan Sistem . . . . .	19
3.3.6. Pengujian Fungsionalitas . . . . .	21
3.3.7. Pengujian <i>Usability</i> . . . . .	21
3.3.8. Analisis <i>Usability</i> . . . . .	21
3.3.9. Penetraion testing . . . . .	22
<b>BAB IV HASIL DAN PEMBAHASAN</b>	<b>23</b>
4.1. ANALISIS KEBUTUHAN . . . . .	23
4.1.1. Kelompok <i>user</i> . . . . .	23
4.1.2. <i>Entity Relationship Diagram</i> . . . . .	23
4.1.3. <i>Use Case Diagram</i> . . . . .	24
4.1.4. <i>Activity Diagram</i> . . . . .	25
4.1.5. <i>Deployment Diagram</i> . . . . .	33
4.2. PEMBUATAN APLIKASI . . . . .	34
4.2.1. Pembuatan <i>back end</i> aplikasi . . . . .	34
4.2.2. Pembuatan <i>front end</i> aplikasi . . . . .	36
4.3. PENGUJIAN APLIKASI . . . . .	50
4.3.1. Pengujian fungsionalitas menggunakan <i>black box testing</i>	50
4.3.2. Pengujian usabilitas menggunakan metode UMUX . . .	54
4.4. ANALISIS USABILITAS ( <i>Usability Analysis</i> ) . . . . .	55
4.5. PENETRATION TESTING MENGGUNAKAN APLIKASI ACUNETIX . . . . .	57
<b>BAB V KESIMPULAN DAN SARAN</b>	<b>58</b>
5.1. KESIMPULAN . . . . .	58
5.2. SARAN . . . . .	58
<b>DAFTAR KEPUSTAKAAN</b>	<b>60</b>
<b>LAMPIRAN 1</b>	<b>61</b>
<b>LAMPIRAN 2</b>	<b>62</b>
<b>LAMPIRAN 3</b>	<b>63</b>

## **DAFTAR TABEL**

Tabel 3.1	Jadwal Penelitian . . . . .	13
Tabel 4.1	REST API . . . . .	35
Tabel 4.2	Pengujian <i>black box</i> . . . . .	50
Tabel 4.3	Pengujian <i>black box</i> admin . . . . .	53
Tabel 4.4	Skenario Pengujian Usabilitas ( <i>Usability Testing</i> ) . . . . .	55
Tabel 4.5	Hasil pengujian dan penilaian UMUX aplikasi Rwikistat . . .	56

## DAFTAR GAMBAR

Gambar 3.1	Diagram Alur Penelitian . . . . .	14
Gambar 3.2	Tampilan <i>Landing Page</i> . . . . .	16
Gambar 3.3	Tampilan <i>Login Page</i> . . . . .	17
Gambar 3.4	Tampilan halaman <i>Modul Belajar</i> . . . . .	17
Gambar 3.5	Tampilan halaman <i>R Compiler</i> . . . . .	18
Gambar 3.6	Tampilan halaman <i>Forum Diskusi</i> . . . . .	18
Gambar 3.7	Tampilan halaman <i>Komentar Forum</i> . . . . .	19
Gambar 3.8	Tahapan Metode SCRUM . . . . .	19
Gambar 4.1	<i>Entity Relationship Diagram</i> . . . . .	24
Gambar 4.2	<i>Use Case Diagram</i> . . . . .	25
Gambar 4.3	<i>Activity Diagram Login</i> . . . . .	26
Gambar 4.4	<i>Activity Diagram R Compiler</i> . . . . .	27
Gambar 4.5	<i>Activity Diagram Pilih Modul</i> . . . . .	28
Gambar 4.6	<i>Activity Diagram</i> tambah dan hapus modull . . . . .	29
Gambar 4.7	<i>Activity Diagram</i> forum diskusi . . . . .	30
Gambar 4.8	<i>Activity Diagram</i> chatbot . . . . .	31
Gambar 4.9	<i>Activity Diagram profile</i> . . . . .	32
Gambar 4.10	<i>Activity Diagram</i> verifikasi <i>user</i> . . . . .	33
Gambar 4.11	<i>Deployment Diagram</i> . . . . .	34
Gambar 4.12	Halaman <i>dashboard</i> . . . . .	37
Gambar 4.13	Halaman <i>login</i> . . . . .	38
Gambar 4.14	Halaman <i>R compiler</i> . . . . .	39
Gambar 4.15	Halaman <i>simpan gambar</i> . . . . .	39
Gambar 4.16	Halaman modul . . . . .	40
Gambar 4.17	Halaman detail modul . . . . .	41
Gambar 4.18	Halaman tambah modul . . . . .	41
Gambar 4.19	Halaman hapus modul . . . . .	42
Gambar 4.20	Halaman forum diskusi . . . . .	43
Gambar 4.21	Halaman detail forum diskusi . . . . .	44
Gambar 4.22	Halaman tambah pertanyaan . . . . .	44
Gambar 4.23	Halaman <i>search</i> . . . . .	45
Gambar 4.24	Halaman <i>chatbot</i> . . . . .	46
Gambar 4.25	Halaman verifikasi <i>user</i> . . . . .	46
Gambar 4.26	Halaman konfirmasi verifikasi <i>user</i> . . . . .	47
Gambar 4.27	Halaman riwayat gambar . . . . .	47

Gambar 4.28 Halaman detail gambar . . . . .	48
Gambar 4.29 Halaman <i>profile user</i> . . . . .	48
Gambar 4.30 Tampilan postingan <i>user</i> . . . . .	49
Gambar 4.31 Tampilan menu <i>bookmark user</i> . . . . .	49
Gambar 4.32 Halaman <i>profile user</i> lain . . . . .	50
Gambar 4.33 Hasil <i>penetration testing</i> . . . . .	57

# BAB I

## PENDAHULUAN

### 1.1. LATAR BELAKANG

Pada era digital saat ini, penggunaan teknologi informasi semakin meluas dan mempengaruhi banyak aspek kehidupan manusia. Dampak dari teknologi informasi dapat dirasakan di setiap bidang yang memungkinkan, salah satunya adalah bidang pendidikan (Raja, 2018). Salah satu bentuk pemanfaatan teknologi yang banyak digunakan dalam pendidikan adalah penggunaan aplikasi dan *website* sebagai media pembelajaran untuk membuat penyajian ilmu pengetahuan lebih mudah dan interaktif.

Pada tahun 2018 beberapa mahasiswa Universitas Syiah Kuala (USK) dibawah naungan Fakultas Matematika dan Ilmu Pengetahuan Alam (FMIPA) telah mengembangkan versi lanjutan dari media pembelajaran yang berupa aplikasi statistik berbasis android yang disebut Rwikistat. Pada aplikasi Rwikistat versi sebelumnya, meskipun berbasis android, pengaturan dan pengisian konten tetap dilakukan melalui *website*. Artinya versi Rwikistat sebelumnya bukan merupakan sebuah aplikasi berbasis android yang sempurna, melainkan hanya berbentuk *web view* (Juwanda, 2018).

Aplikasi Rwikistat memiliki fitur yang sangat bermanfaat bagi mahasiswa atau siapapun yang ingin belajar statistika, namun saat ini sudah ditinggalkan karena penggunaanya yang tidak praktis. Berdasarkan hal ini, aplikasi Rwikistat yang sebelumnya hanya berupa *web view* dikembangkan kembali dengan versi *native mobile* untuk dapat digunakan dalam *platform mobile* baik berbasis android maupun iOS agar penggunaannya lebih praktis dan efisien.

Di aplikasi Rwikistat versi sebelumnya terdapat 3 fitur antara lain penyajian teori statistika, penyajian berita-berita terkini tentang ilmu statistika, dan fitur *R compiler*. Pada pengembangan kali ini fitur berita dihilangkan karena dirasa kurang perlu dan diganti dengan fitur forum diskusi sehingga fitur yang akan dikembangkan yaitu forum diskusi, penyajian teori dan *R compiler*. Fitur *R compiler* yang dikembangkan akan menampilkan grafik interaktif dari soal-soal statistika yang di *input*. Meskipun fitur yang akan dikembangkan ada 3, penulis nantinya hanya akan mengembangkan forum diskusi saja karena 2 fitur lainnya sudah dikembangkan terlebih dahulu oleh rekan penulis.

Forum diskusi merupakan fitur yang diperlukan dalam pengembangan aplikasi Rwikistat karena memberikan pengguna sebuah wadah untuk berinteraksi, bertukar informasi, dan mendiskusikan berbagai topik terkait penggunaan aplikasi maupun konsep-konsep statistika. Dalam konteks penggunaan aplikasi, forum

diskusi memungkinkan pengguna untuk saling berbagi tips, trik, dan pengalaman dalam menggunakan fitur-fitur aplikasi Rwikistat, sehingga dapat meningkatkan pemahaman dan efektivitas penggunaan aplikasi.

Selain itu, forum diskusi juga memberikan kesempatan bagi pengguna untuk bertanya dan berdiskusi tentang konsep-konsep statistika yang sulit dipahami atau membingungkan. Diskusi-diskusi ini dapat melibatkan pengguna lain yang memiliki pemahaman yang lebih baik atau mungkin dosen dan ahli statistika yang terlibat dalam pengembangan aplikasi. Dengan adanya forum diskusi, pengguna dapat saling membantu dan mendukung satu sama lain dalam memahami materi statistika yang kompleks.

Lebih jauh lagi, forum diskusi dapat menjadi sarana untuk membangun komunitas yang solid di sekitar pengguna aplikasi Rwikistat. Dalam komunitas ini, pengguna dapat saling bertukar ide, berkolaborasi dalam proyek-proyek statistika, atau bahkan mengadakan acara-acara atau pertemuan yang berkaitan dengan bidang statistika. Dengan demikian, forum diskusi tidak hanya berfungsi sebagai alat bantu dalam penggunaan aplikasi, tetapi juga sebagai tempat untuk membangun jejaring dan komunitas yang kuat dalam dunia statistika.

Selain dari semua fitur yang sudah dijelaskan sebelumnya, penulis juga akan mengembangkan *Representational State Transfer Application Programming Interface* atau yang biasa disebut REST API untuk membantu pengembangan Rwikistat versi *mobile*. Aplikasi *mobile* membutuhkan sebuah API sebagai media penghubung antara aplikasi *mobile* dan server. API dikembangkan menggunakan *framework* NextJS.

## 1.2. RUMUSAN MASALAH

Berkaca dari latar belakang, maka terdapat beberapa masalah yang dapat dirumuskan sebagai berikut:

1. Bagaimana mengembangkan forum diskusi yang dapat memenuhi kebutuhan pengguna pada aplikasi Rwikistat?
2. Bagaimana membangun REST API untuk website dan aplikasi *mobile* Rwikistat?
3. Bagaimana cara menguji aplikasi Rwikistat yang dikembangkan?

## 1.3. TUJUAN PENELITIAN

Tujuan dari penelitian ini adalah sebagai berikut:

1. Mengembangkan forum diskusi pada aplikasi Rwikistat yang dapat memenuhi kebutuhan pengguna.

2. Membangun REST API untuk website dan aplikasi *mobile* Rwikistat.
3. Melakukan pengujian fungsionalitas dan usabilitas pada aplikasi Rwikistat yang dikembangkan.

#### **1.4. MANFAAT PENELITIAN**

Adapun manfaat dari penelitian ini adalah:

1. Meningkatkan partisipasi pengguna dalam berdiskusi dan berinteraksi dalam konteks pembelajaran statistik dan data science pada aplikasi Rwikistat yang dikembangkan..
2. Memberikan kontribusi terhadap penelitian dan pengembangan aplikasi pembelajaran statistik dan data science di masa depan.
3. Memberikan pengalaman dan keterampilan dalam melakukan penelitian, analisis data, dan pengembangan aplikasi yang dapat berguna untuk karir di bidang pembelajaran statistik dan data science.

## **BAB II**

### **TINJAUAN KEPUSTAKAAN**

#### **2.1. FORUM DISKUSI *ONLINE***

Komunikasi dan partisipasi memiliki peran penting dalam meningkatkan kualitas pengajaran dan pembelajaran. Salah satu alasan mengapa penggunaan internet dalam proses belajar mengajar semakin meningkat adalah karena teks memungkinkan orang untuk terlibat secara lebih dalam dengan informasi yang sedang berlangsung. Pendidikan *online* memungkinkan penggunaan teknologi instruksional yang adaptif, terkoordinasi, dan kolaboratif, yang memperluas cara pertukaran ide, pelaksanaan penelitian, dan penyajian penemuan individu dapat dilakukan dengan cara yang nyaman bagi siswa (Mtshali, 2020).

Dalam konteks ini, forum diskusi *online* menjadi salah satu alat komunikasi yang dapat digunakan dengan bantuan komputer. Forum diskusi *online* merupakan aplikasi berbasis web yang banyak digunakan untuk menghubungkan orang-orang dengan minat yang sama. Dalam pengajaran di kelas besar, forum ini menjadi alat yang berguna untuk memfasilitasi komunikasi. Kelebihannya terletak pada efektivitas konseling dan kerjasama antara guru dan siswa, serta antar siswa. Forum diskusi *online* memungkinkan komunikasi asinkron yang mendukung interaksi dan kolaborasi sosial tanpa harus mempertimbangkan waktu komunikasi seperti yang terjadi dalam ruang obrolan *online*. Keberadaan forum ini sebagai wadah diskusi kapan saja juga menjadi insentif yang kuat bagi mereka yang berencana untuk mendaftar dalam modul *online*.

Forum diskusi *online* dapat memberikan manfaat bagi pembelajaran siswa dalam berbagai konteks. Penggunaan forum diskusi *online* dalam pengajaran dapat meningkatkan partisipasi siswa, interaksi mereka, dan kualitas komunikasi yang terjalin. Selain itu, forum diskusi *online* juga memungkinkan adanya interaksi di luar jam pelajaran antara siswa dengan sesama siswa maupun dengan guru (Galletly, 2020).

Dengan demikian, forum diskusi *online* menjadi ruang yang efektif untuk memfasilitasi diskusi dan kolaborasi yang lebih dalam antara peserta didik. Melalui forum ini, mereka dapat saling bertukar pendapat, memberikan umpan balik, dan berbagi penemuan atau penelitian mereka secara *real-time*. Hal ini mendorong siswa untuk terlibat secara aktif dalam proses pembelajaran dan mengembangkan kemampuan komunikasi mereka dengan lebih baik.

Selain itu, forum diskusi *online* juga mengatasi keterbatasan waktu dan ruang yang sering terjadi dalam proses belajar mengajar konvensional. Melalui forum ini, siswa dapat berinteraksi dengan sesama siswa dan guru secara fleksibel, tanpa adanya

batasan waktu tertentu. Hal ini memberikan kesempatan bagi siswa yang memiliki keterbatasan waktu atau jarak untuk tetap terlibat dalam diskusi dan mendapatkan bimbingan dari guru.

## 2.2. WEBSITE

*Website* merupakan suatu alat yang digunakan untuk mengkomunikasikan informasi kepada khalayak umum melalui jaringan internet. Penggunaan *website* memberikan sejumlah keuntungan, di antaranya adalah kemudahan akses terhadap informasi, distribusi informasi yang efisien, dan fleksibilitas akses melalui berbagai *platform*. Kelebihan-kelebihan ini menjadi alasan utama mengapa banyak individu dan organisasi memilih *website* sebagai media untuk menyampaikan informasi kepada publik.

Salah satu keunggulan utama *website* adalah kemudahan akses terhadap informasi. Dengan adanya *website*, pengguna dapat dengan cepat dan mudah mencari informasi yang mereka butuhkan. Dalam era digital saat ini, website telah menjadi sumber informasi yang dapat diakses oleh siapa saja, kapan saja, dan di mana saja. Hal ini memungkinkan penyebaran informasi yang lebih luas dan merata, sehingga masyarakat dapat memperoleh informasi yang relevan dan *up-to-date* dengan mudah (Rerung, 2020).

### 2.2.1. Framework

*Framework* merupakan kumpulan komponen yang bekerja sama untuk menciptakan sebuah arsitektur desain yang dapat digunakan secara berulang dalam pembangunan suatu hal. Dalam ilmu perangkat lunak, *Framework* adalah sebuah *platform* konkret dan terstruktur yang berisi kode-kode dengan fungsionalitas generik yang dapat di-*override* secara selektif oleh pengembang atau pengguna. *Framework* mengadopsi bentuk dari sebuah *library*, di mana *application program interface* (API) dapat digunakan berulang-ulang di berbagai tempat dan waktu, bahkan saat perangkat lunak sedang dikembangkan. Beberapa fitur yang membedakan *Framework* dengan bentuk *library* lainnya adalah sebagai berikut:

1. *Default Behavior*: Sebelum disesuaikan dengan kebutuhan, sebuah framework memiliki perilaku tertentu yang khas sesuai dengan aksi pengguna.
2. *Inversion of Control*: Berbeda dengan perpustakaan, kontrol aliran aplikasi dalam sebuah framework diatur oleh framework itu sendiri, bukan oleh fungsi pemanggil.
3. *Extendibility*: Pengguna dapat memperluas (extend) sebuah framework dengan mengganti kode bawaan dengan kode yang dibuat pengguna.

4. *Non-modifiable Framework Code*: Pengguna dapat memperluas suatu *Framework*, tetapi tidak dapat memodifikasi keseluruhan kode *Framework*.

Tujuan dari *Framework* perangkat lunak adalah untuk menyederhanakan lingkungan pengembangan, sehingga pengembang dapat fokus pada kebutuhan proyek daripada harus menghadapi kerumitan dalam menggunakan fungsi dan *library* yang berulang-ulang. Dengan adanya *Framework*, pengembangan dengan sintaks yang kompleks dapat diatasi dengan mudah. Dahulu, pengembangan didasarkan pada rangkaian baris kode, namun sekarang pengembangan didasarkan pada pengiriman (*delivery*) (Edwin, 2018).

### 2.3. REACT.JS

*React* adalah sebuah *library JavaScript* untuk membangun *interface* pengguna. *React* dibangun pada tahun 2011 oleh Jordan Walke, seorang insinyur perangkat lunak di *Facebook*. *library* ini dibuka sebagai *open source* pada tahun 2013 dan terus mengalami penambahan fitur sejak saat itu. Pada tahun 2015, *React Native*, sebuah *library* untuk pengembangan aplikasi *mobile*, juga di*open sourcekan*, dan pada tahun 2017, pustaka untuk pengembangan *virtual reality*, *React360*, dirilis. Saat ini, *React* dikembangkan oleh *Facebook* dan komunitas pengguna individu. *Facebook* menggunakan *React* dalam proyek-proyeknya, seperti *Facebook* dan *Instagram*. *React* dirancang untuk meningkatkan pengembangan *interface* pengguna yang interaktif dengan membuat pembaruan tampilan menjadi lebih mudah ketika data berubah. Hal ini dilakukan dengan membagi tampilan menjadi komponen-komponen kecil yang dapat digabungkan untuk menciptakan antarmuka pengguna yang kompleks. Komponen-komponen ini dibangun menggunakan bahasa *JavaScript* daripada templat, sehingga memungkinkan aliran data yang mudah (Saks, 2019).

*React* adalah sebuah *library JavaScript* deklaratif, efisien, dan fleksibel yang ditujukan untuk membangun antarmuka pengguna (*UI*) yang interaktif dan membangun komponen *UI* yang dapat digunakan kembali. *React* menggunakan pendekatan berbasis komponen, yang berarti ia terdiri dari bagian-bagian kecil yang independen yang disebut komponen, sehingga membuat kode lebih mudah ditangani dan lebih dapat diprediksi. Karena manipulasi *DOM* adalah tugas yang berat dan mempengaruhi kinerja halaman *web*, *virtual DOM* menjadi solusi yang digunakan. Perubahan pertama-tama dilakukan pada *virtual DOM*, kemudian perbedaan antara *DOM* aktual dan *virtual* dihitung. Selanjutnya, *node-node* yang mengalami perubahan akan dirender ulang. Dengan cara ini, menghitung objek *JS* sederhana di memori jauh lebih cepat daripada memanipulasi objek *DOM* untuk setiap transformasi (Jartarghar, 2022).

## 2.4. TYPESCRIPT

*TypeScript* adalah *superset* dari bahasa *JavaScript* yang berfokus pada menghasilkan kode yang aman dan dapat diprediksi yang dapat dieksekusi oleh setiap *runtime JavaScript*. Fitur utamanya adalah tipe statis, yang membuat pekerjaan dengan *JavaScript* lebih dapat diprediksi bagi para programmer yang sudah terbiasa dengan bahasa seperti *C-Sharp* dan *Java*. Fitur unggulan *TypeScript* difokuskan pada produktivitas pengembang, terutama melalui penggunaan tipe statis yang membantu membuat sistem tipe *JavaScript* lebih mudah digunakan. Fitur produktivitas lainnya, seperti kata kunci pengendalian akses dan sintaksis konstruktor kelas yang ringkas, membantu mencegah kesalahan pemrograman umum (Freeman, 2021).

*JavaScript* adalah bahasa yang dinamis, yang berarti, antara lain, bahwa tidak ada tipe untuk variabel-variabelnya. Dan saya tahu apa yang akan Anda katakan, anda memiliki beberapa tipe dasar seperti *Number*, *Object*, atau *String*, tetapi tidak ada pengecekan tipe statis yang terjadi pada setiap saat tertentu, anda dapat menulis kode dari daftar 2-1 tanpa ada yang rusak. Hingga saat ini, cara kerja dengan *TypeScript* adalah dengan menyiapkan proses *build* menggunakan alat otomatisasi seperti *webpack*, atau, di masa lalu, *Gulp* atau *Grunt*. Alat tersebut memungkinkan anda membuat proses untuk mentranspilasi kode Anda (dengan kata lain, menerjemahkannya menjadi *JavaScript*) sebelum dapat dieksekusi. Ini adalah tugas yang sangat umum sehingga sudah ada alat-alat yang secara otomatis mengonfigurasi proses tersebut untuk Anda saat Anda mulai proyek baru. Misalnya, ambil aplikasi *create-react-app* yang dirancang untuk membantu Anda membuat proyek *React* baru, jika Anda memilih untuk menggunakannya dengan *TypeScript*, aplikasi tersebut akan mengatur langkah transpilasi tersebut untuk Anda (Doglio, 2020).

## 2.5. OPENAI

*OpenAI*, sebuah perusahaan riset kecerdasan buatan terkemuka, telah menjadi pionir dalam mengembangkan model-model *AI* inovatif seperti *ChatGPT* dan *Codex*. Model-model ini telah membawa kemajuan signifikan dalam pemrosesan bahasa alami dan pemahaman konteks secara otomatis (Roumeliotis, 2023). Namun, inovasi *OpenAI* tidak berhenti di situ. Perusahaan ini juga memperluas jangkauan teknologinya dengan menyediakan API yang memungkinkan pengguna untuk melakukan *fine-tuning* pada model-model canggih seperti GPT-3.

Melalui API ini, pengguna dapat menyesuaikan model transformer untuk tugas-tugas tertentu, memperbaiki kinerja model dalam menangani tugas-tugas spesifik dengan lebih efisien dan akurat. Dengan demikian, *OpenAI* tidak hanya

menciptakan model-model *AI* yang revolusioner, tetapi juga memberdayakan pengguna untuk mengoptimalkan penggunaannya dalam berbagai konteks dan aplikasi. Dengan kemampuan fine-tuning ini, *OpenAI* membuka pintu bagi eksplorasi lebih lanjut dalam pengembangan teknologi *AI*, memperluas potensi penerapannya dalam pemrosesan bahasa alami dan bidang-bidang lainnya, dan secara luas memberikan manfaat yang signifikan bagi pengguna di seluruh dunia (Sun, 2023).

## 2.6. PENETRATION TESTING

*Penetration testing* adalah proses penting dalam upaya untuk meningkatkan keamanan sistem atau jaringan. Dalam praktiknya, para profesional keamanan sering menggunakan metode yang umumnya digunakan oleh peretas (*hacker*) untuk mengidentifikasi dan mengeksloitasi kerentanan keamanan yang ada (Sandhya, 2017). Salah satu alat yang sering digunakan dalam melakukan *penetration testing* adalah *Acunetix*, sebuah pemindai kerentanan web otomatis yang sangat andal. Dengan menggunakan *Acunetix*, pemilik *website* dapat melakukan *penetration testing* secara menyeluruh pada situs web, mengidentifikasi celah keamanan potensial, dan menganalisis risiko yang terkait dengan kerentanan tersebut (Rajan, 2017). Melalui pendekatan yang terstruktur dan sistematis, saya dapat memastikan bahwa situs web tersebut terlindungi secara optimal dari serangan peretas dan kerentanan keamanan lainnya. Dengan demikian, penggunaan *Acunetix* dalam praktik *penetration testing* menjadi salah satu strategi yang efektif dalam menjaga keamanan sistem dan jaringan.

## 2.7. SCRUM

*Scrum* adalah salah satu praktik *agile*, ini adalah pendekatan inkremental dan iteratif yang digunakan untuk mengelola pekerjaan kompleks, yaitu mengembangkan produk perangkat lunak kompleks dengan persyaratan bisnis yang sering berubah. *Scrum* terdiri dari peran-peran yang telah ditentukan sebelumnya dan juga memiliki sekelompok proses. Peran-peran dalam *Scrum* termasuk:

- *SCRUM Master*: Bertanggung jawab menjalankan proses *Scrum*.
- Pemilik Produk: Pemangku kepentingan atau pelanggan.
- Tim *Scrum*: Terdiri dari 7 orang. Produk secara keseluruhan dibagi menjadi peningkatan kecil yang merupakan *deliverable* yang dapat dikirimkan dan diperiksa pada akhir setiap *sprint*.

Penerapan *Scrum* dilakukan melalui upacara atau pertemuan. Upacara *Scrum* utama adalah pertemuan *sprint*, *daily scrum*, *sprint review*, dan *sprint retrospective*.

*Sprint* adalah periode kecil yang dibatasi oleh waktu yang diperlukan oleh tim untuk menyelesaikan sekelompok tugas dalam periode tersebut. Pertemuan perencanaan rilis digunakan untuk merencanakan dan memperkirakan serangkaian *sprint*, pertemuan ini bersifat opsional. *Scrum* memiliki beberapa tahap untuk dijalankan, tahap-tahap *Scrum* adalah sebagai berikut:

1. *Sprint*: Mengontrol mekanisme tarik yang membatasi beban kerja yang masuk.
2. Perencanaan *Sprint*: Memungkinkan penilaian diri setiap tugas yang memulai kejelasan dan pertumbuhan sehubungan dengan DoD (*Definition of Done*).
3. *Daily Scrum*: Meningkatkan koordinasi di antara tim untuk memperlihatkan dan mengungkapkan ide-ide mereka.
4. *Sprint Review*: Melibatkan tim *scrum* bersama pemangku kepentingan lainnya. Mengambil umpan balik dari tim scrum dan dilakukan pemeriksaan di sini.
5. *Sprint Retrospective*: Memberikan kesempatan untuk mengevaluasi hambatan-hambatan saat ini dan menerima umpan balik untuk *sprint* berikutnya.

Sebuah kelompok fitur dalam *product backlog* dijelaskan dalam *sprint* (daftar prioritas kebutuhan tingkat tinggi pelanggan). Identifikasi *backlog* ini dilakukan selama konferensi perencanaan *sprint*. Tim bekerja untuk *sprint* saat ini. Tidak ada yang diizinkan untuk mengubah tugas *backlog* saat pertemuan sedang berlangsung. Setelah berakhirnya *sprint*, tim akan memperlihatkan penggunaan perangkat lunak yang lebih canggih.

*Scrum* juga memiliki beberapa keunggulan, yaitu sebagai berikut;

1. *Scrum* mendukung tim untuk menyelesaikan produk mereka dengan cepat dan konsisten.
2. *Scrum* menjamin penggunaan uang dan waktu yang efisien.
3. Proyek-proyek besar dibagi menjadi *sprint* yang dapat dipelihara dengan cepat.
4. Produk yang dikembangkan dikodekan dan diuji sepanjang analisis *sprint*.
5. Setiap tugas dilakukan dengan baik untuk pengembangan perangkat lunak yang cepat.
6. Anggota tim dapat memperoleh pemahaman yang jelas melalui pertemuan *scrum*.
7. Mendapatkan umpan balik dari pelanggan dan pemangku kepentingan untuk perbaikan yang lebih baik.

- Upaya mandiri dari setiap anggota tim terlihat dalam setiap pertemuan *scrum* (Hema, 2020).

## **2.8. BLACKBOX TESTING**

Pengujian *black box* adalah pengujian yang hanya menguji bagian luar perangkat lunak. Pengujian *black box* adalah teknik yang berfokus pada persyaratan fungsional perangkat lunak berdasarkan spesifikasi persyaratan perangkat lunak. Terdapat beberapa metode pengujian dalam pengujian *black box*, seperti partisi ekuivalen, analisis nilai batas, grafik penyebab-akibat, pengujian perbandingan, pemilihan data acak, pengujian fitur, pengujian semua pasangan, *fuzzing*, pengujian matriks ortogonal, pengujian sampel, pengujian kekokohan, pengujian perilaku, pengujian kinerja, pengujian ketahanan, dan lain-lain (Santi, 2022).

Istilah "*black-box*" biasanya digunakan untuk menjelaskan semua model pembelajaran mesin yang (dari segi matematika) sangat sulit dijelaskan dan dipahami oleh para ahli di bidang praktis. Model-model berbasis *black-box* ini dapat dikelompokkan menjadi kategori-kategori berikut: berbasis *hyperplane*, seperti yang digunakan oleh *Support Vector Machine* (SVM), terinspirasi oleh jaringan saraf biologis yang membentuk otak hewan berbasis logika probabilitas dan kombinatorik, seperti *Probabilistic Logic Networks* (PLN), dan yang berbasis pada contoh (dikenal sebagai pembelajaran malas) di mana fungsi hanya diperkirakan secara lokal, seperti *k-nearest neighbors*. Di bawah ini, kami akan menjelaskan model-model yang paling sering digunakan dalam setiap kategori ini dan mengapa mereka disebut sebagai model *black-box* (Loyola-Gonzalez, 2019).

## **2.9. USABILITY TESTING**

*Usability Testing* adalah metode pengujian yang difokuskan pada pengguna saat mereka menjalankan tugas atau menjelaskan tujuan tertentu yang diamati. Proses pengujian ketergunaan mencakup perancangan dan persiapan pengujian, pelaksanaan sesi pengujian, analisis hasil pengujian, dan komunikasi hasil pengujian. Aspek yang menjadi fokus pengukuran dalam pengujian ketergunaan adalah efektivitas, efisiensi, dan kepuasan. Efektivitas diukur berdasarkan sejauh mana pengguna dapat menyelesaikan tugas dalam batas waktu maksimal 15 menit. Efisiensi diukur dengan rata-rata waktu yang dibutuhkan pengguna untuk menyelesaikan tugas, dengan catatan bahwa waktu maksimal 15 menit digunakan untuk tugas yang tidak selesai. Kepuasan diukur menggunakan metrik positif dalam sistem *User Metric for User Experience* (UMUX), dan pengguna diminta untuk memilih sistem favorit setelah pengujian keseluruhan sistem dilakukan (Riihiaho, 2018).

*Usability Testing* menempatkan fokus pada pengguna, oleh karena itu penting untuk melakukan seleksi yang hati-hati terhadap pengguna yang akan menjadi penguji. Idealnya, penguji tidak dikenal sebelumnya untuk menghindari penilaian yang bersifat subyektif. Selain itu, penguji juga diharapkan memiliki keterkaitan dan relevansi dengan aplikasi yang sedang diuji. Lokasi dan jenis pertanyaan yang diajukan juga memainkan peran penting dalam pengumpulan data hasil pengujian yang akan menjadi dasar untuk pembaharuan aplikasi yang sedang dikembangkan. Pada intinya, karena pengujian ini berfokus pada pengguna, mereka diminta untuk berpikir secara luas dan memberikan ulasan yang positif maupun negatif kepada tim pengembang (Barnum, 2020).

## **2.10. USABILITY METRICS FOR USER EXPERIENCE (UMUX)**

*Usability Metric for User Experience* (UMUX), juga dikenal sebagai Metrik Pengguna untuk Pengalaman Pengguna, adalah metode pengujian ketergunaan yang menggunakan skala *Likert* sebanyak 4 pertanyaan sebagai penilaian objektif dan subjektif terhadap penggunaan sistem aplikasi yang sedang diuji. UMUX umumnya digunakan bersama dengan *System Usability Scale* (SUS) untuk memperoleh pengukuran penggunaan yang konsisten. Meskipun keduanya memiliki skala dan kuesioner yang disusun, UMUX memiliki 4 kuesioner dan 7 skala jawaban dari responden, sedangkan SUS memiliki 10 kuesioner dengan 7 skala jawaban dari responden.

Beberapa peneliti seperti Finstad (2010), Berkman dan Karahoca (2016), Borsci (2015), dan Lewis (2015) telah mengajukan beberapa properti psikometrik yang diinginkan dalam UMUX, termasuk tingkat keandalan (koefisien alfa lebih dari 0,80), validitas kebersamaan (korelasi dengan SUS lebih besar dari 0,55 dan korelasi dengan CSUQ sama dengan -0,65), serta sensitivitas untuk membedakan variasi tingkat dari variabel independen. Seperti halnya CSUQ dan SUS, UMUX dan pengukuran yang terkandung di dalamnya tersedia untuk digunakan oleh para peneliti tanpa perlu membayar lisensi (Lewis, 2018).

## **2.11. PENELITIAN TERKAIT**

Pada tahun 2010, diluncurkan versi pertama dari *Rwikistat*, sebuah platform pembelajaran statistik yang interaktif dan komunikatif. Rwikistat menggabungkan perangkat lunak statistik *R* dengan aplikasi wiki untuk memperkuat bidang pengajaran statistik melalui media pendidikan interaktif. *Platform* ini dapat diakses melalui intranet atau internet dan cocok digunakan dalam pengajaran statistik dasar di universitas. *Rwikistat* menghasilkan situs web pembelajaran *e-learning*, *CD/DVD* interaktif untuk pembelajaran statistik, dan konten statistik dasar seperti statistik

fundamental, distribusi frekuensi, kombinatorik, variabel acak, distribusi probabilitas, dan teori pengambilan sampel. Konten RWikiStat disusun dalam urutan berikut: teori, aplikasi, pengembangan (studi kasus), praktik masalah/pemecahan masalah, dan contoh-contoh yang dijalankan dengan menggunakan sistem Rweb (Hizir, 2010).

Pada tahun 2012, dikembangkanlah *RWikiStat 2.0*, sebuah sistem pembelajaran statistik berbasis web yang menggabungkan teknologi Wiki dengan perangkat lunak statistik *R*. Sistem ini merupakan sumber terbuka yang terus dikembangkan secara bertahap di bawah sistem operasi *Linux*. *RWikiStat 2.0* dirancang untuk memberikan proses pembelajaran interaktif bagi mahasiswa yang menghadapi kesulitan dalam mempelajari statistik, terutama pada tahun pertama kuliah. Pengunjung dapat dengan mudah memodifikasi konten web, seperti menambahkan, menghapus, atau mengubah artikel. Sistem ini dapat digunakan secara offline atau individu, dan juga dapat digunakan secara bersama-sama dengan berbagi file tanpa perlu instalasi sebelumnya, yang memakan waktu dan ruang di *hard disk*. Pengembangan *Live DVD* juga memastikan keamanan aplikasi tanpa kehilangan data yang disebabkan oleh kerusakan pada media *DVD*. Namun, saat menggunakan *Live DVD*, data yang disimpan tidak dapat diubah, ditambahkan, atau dihapus. Pembaruan perlu dilakukan jika terdapat perubahan data atau formula dalam *RWikiStat 2.0* (Hizir, 2012).

Pada tahun 2019, pengembangan *RWikiStat* dilanjutkan dengan meluncurkan *RWikiStat 3.0*, sebuah aplikasi Android untuk pembelajaran statistik berbasis RWeb dan Wiki. *RWikiStat 3.0* merupakan pengembangan dari versi sebelumnya, *RWikiStat 2.0*, dan memiliki antarmuka yang responsif dan ramah pengguna untuk mempelajari analisis statistik menggunakan bahasa pemrograman *R*. Evaluasi kepuasan pengguna terhadap aplikasi ini dilakukan melalui survei yang dilakukan setelah pelatihan di tiga universitas di Banda Aceh, Indonesia. Hasil survei menunjukkan bahwa responden secara keseluruhan merasa puas dengan aplikasi ini, terutama dengan antarmuka, responsifitas, dan kegunaannya. Responden juga memiliki niat yang tinggi untuk merekomendasikan aplikasi ini kepada teman, rekan kerja, atau orang lain. Namun, beberapa responden memberikan saran untuk perbaikan pada baris perintah dan antarmuka aplikasi ini (Hizir, 2019).

## **BAB III**

### **METODOLOGI PENELITIAN**

#### **3.1. WAKTU DAN LOKASI PENELITIAN**

Penelitian ini dilakukan terhitung dari bulan Mei 2023 sampai dengan bulan Januari 2024 yang bertempat di gedung A Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Syiah Kuala. Kurang lebih dalam pelaksanaan penelitian ini memakan waktu 6 bulan dari awal penulisan, pengembangan, sampai dengan penulisan Laporan Akhir.

No	Keterangan	Bulan di 2023 dan 2024								
		Mei	Jun	Jul	Ags	Sep	Okt	Nov	Des	Jan
1	Studi Literatur									
2	Penulisan Proposal									
3	Pengembangan Apikasi									
4	Pengujian Sistem									
5	Penulisan Laporan Akhir									

**Tabel 3.1.** Jadwal Penelitian

#### **3.2. ALAT DAN BAHAN**

Berikut ini adalah alat dan bahan yang digunakan oleh penulis dalam mengerjakan Tugas Akhir ini.

##### **3.2.1. Perangkat keras (*Hardware*)**

Komponen perangkat keras yang digunakan pada penelitian ini adalah 1 unit laptop merk HP LAPTOP 14s dengan Prosesor Intel® Core™ i3- 7020U (2.230 GHz) dan Memori 8 GB DDR4 RAM, Penyimpanan 1TB SSHD

##### **3.2.2. Perangkat lunak (*Software*)**

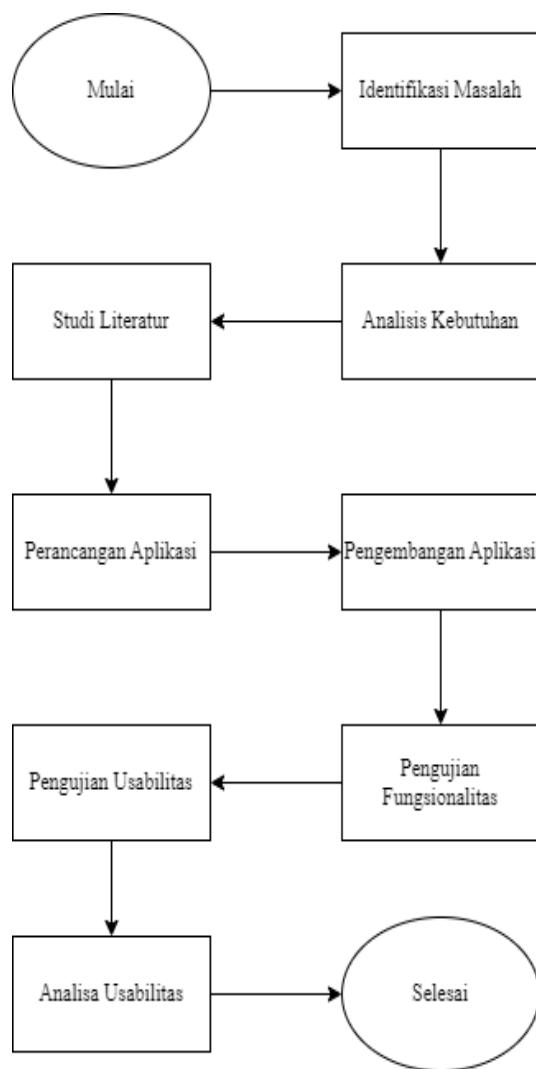
Komponen perangkat lunak yang digunakan pada penelitian ini adalah sebagai berikut:

- Sistem Operasi Windows 10
- *Opera Browser*
- *Visual Studio Code*

- *nodeJs* versi 16.13.2
- *React* versi 18.2.0
- *Postman* versi 10.13
- *Acunetix* versi 10.5

### 3.3. METODE PENELITIAN

Metode penelitian yang dilakukan pada penelitian terdiri daripada beberapa tahapan sebagaimana ditunjukkan pada gambar diagram 3.1 berikut ini.



**Gambar 3.1.** Diagram Alur Penelitian

#### 3.3.1. Identifikasi Masalah

Identifikasi masalah merupakan tahapan yang dilakukan penulis untuk mengidentifikasi masalah yang dialami sehingga penelitian ini perlu dilakukan.

Masalah yang teridentifikasi oleh penulis adalah pada versi aplikasi Rwikistat yang telah dikembangkan baru-baru ini, *user* tidak bisa berkomunikasi dengan *user* lainnya. Oleh karena itu, fitur forum diskusi perlu dikembangkan guna untuk memudahkan *user* menggunakan aplikasi Rwikistat nantinya.

### 3.3.2. Studi Literatur

Studi literatur merupakan proses pencarian bahan referensi selama proses tersebut untuk mempelajari Studi literatur berlangsung melalui pengolahan bahan penelitian Membaca, mempelajari, mengumpulkan teori-teori yang berhubungan dengan penelitian baik dengan browsing *website*, buku dan majalah yang berhubungan dengan penelitian majalah nasional dan internasional, buku terbitan dan Halaman internet tentang masalah yang diperiksa di untuk mempelajari Tinjauan literatur dapat dikembangkan untuk mengatasi kesenjangan dari penelitian sebelumnya. Langkah-langkah Studi literatur menjadi lebih mudah lakukan latar belakang, rumuskan masalah, tujuan dan konfirmasi ulasan literatur dalam penelitian.

### 3.3.3. Analisis Kebutuhan

Tahap analisis kebutuhan adalah tahapan yang dilakukan dengan tujuan untuk mengetahui kebutuhan berdasarkan masalah yang telah diidentifikasi sebelumnya guna membantu pengembangan aplikasi. Kebutuhan dari sistem yang dikembangkan adalah sebagai berikut:

1. **Kebutuhan fungsional** adalah seperangkat fitur dan kemampuan yang harus dimiliki oleh sistem untuk memenuhi kebutuhan *user*. Kebutuhan fungsional yang diperoleh dari identifikasi masalah yang telah dilakukan adalah sebagai berikut:
  - Terdapat fitur yang digunakan untuk memberikan pertanyaan seputar aplikasi maupun statistika .
  - Terdapat halaman yang menampilkan pertanyaan-pertanyaan yang tersedia dan memiliki kolom yang digunakan untuk menjawab pertanyaan.
  - *user* bisa bertanya langsung kepada para *expert* di bidang-bidang tertentu agar jawaban yang diterima bisa lebih dipercaya.
  - Terdapat fitur *search* yang bisa digunakan untuk mencari pertanyaan yang spesifik.
  - terdapat halaman yang menampilkan daftar *user* sehingga *user-user* lain dapat mengetahui siapa saja yang *expert* di bidangnya.

2. **Kebutuhan non-fungsional** adalah kebutuhan yang dibutuhkan sistem agar bisa berjalan secara optimal. Kebutuhan-kebutuhan tersebut antara lain sebagai berikut:

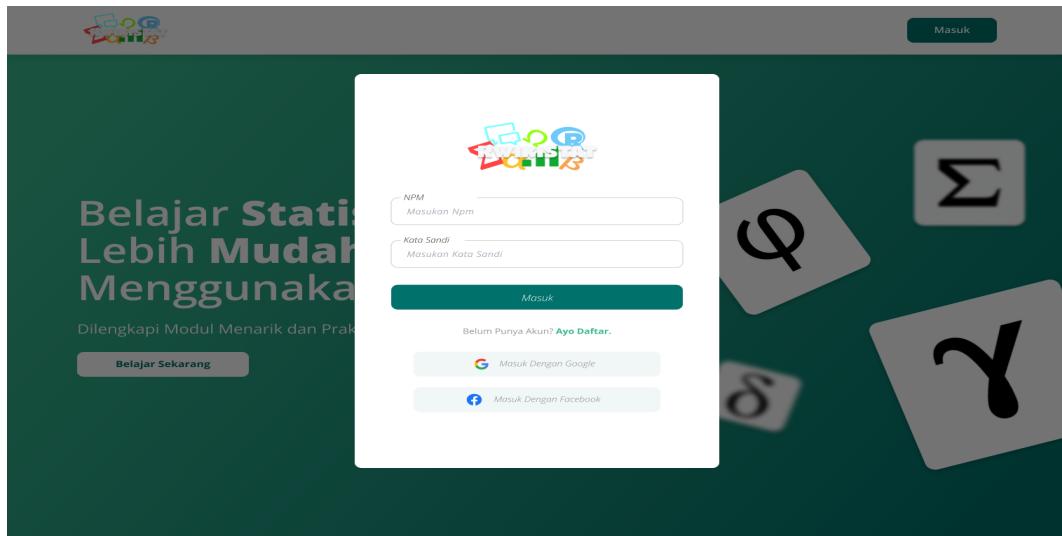
- Sistem memiliki tampilan yang simpel dan mudah dipahami.
- Dapat dijalankan di semua jenis *browser*.

### 3.3.4. Perancangan Sistem

Perancangan sistem merupakan tahapan yang bertujuan untuk agar penelitian dapat dikerjakan dengan jelas, baik dan terstruktur. Dalam tahap ini dilakukannya desain prototipe dari aplikasi berdasarkan analisa kebutuhan pada tahapan sebelumnya. Desain prototipe dari aplikasi (Rwikistat) dapat dilihat dari Gambar 3.2 sampai Gambar 3.7.



Gambar 3.2. Tampilan *Landing Page*



**Gambar 3.3.** Tampilan *Login Page*

**Gambar 3.4.** Tampilan halaman *Modul Belajar*

The screenshot shows a web-based R compiler interface. On the left, there's a sidebar with links for 'Modul Belajar', 'R Compiler' (which is currently selected), and 'Forum Diskusi'. The main area has a search bar at the top. Below it, there are two panes: 'testing.r' on the left containing R code and 'Output' on the right showing the results. The 'testing.r' pane contains the following code:

```

1 "Hello World!"
2   5 + 5
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21

```

The 'Output' pane shows the results of running the code:

```

Hello World!
10

```

At the bottom of the main area, there are 'Upload File' and 'Run' buttons.

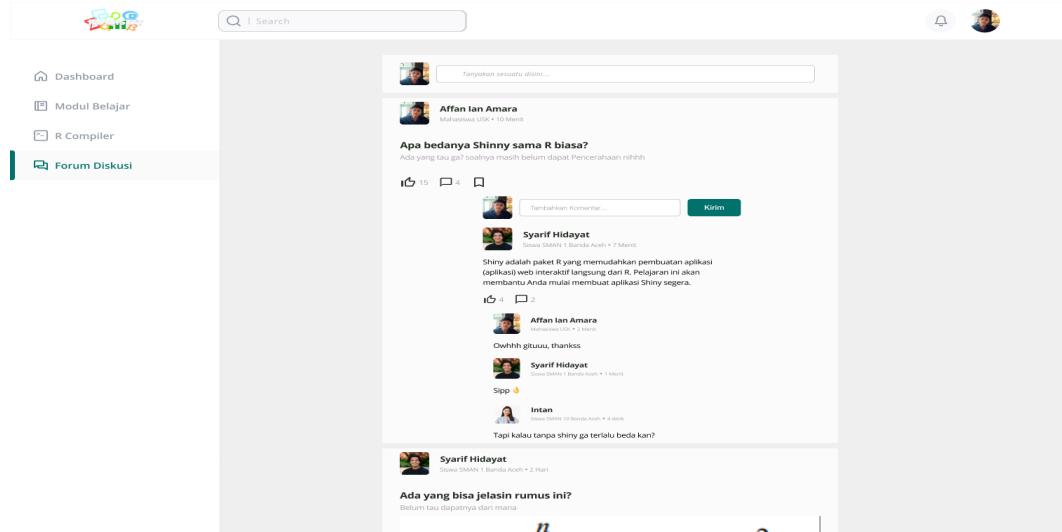
**Gambar 3.5.** Tampilan halaman *R Compiler*

The screenshot shows a forum discussion page. The sidebar on the left includes 'Modul Belajar', 'R Compiler' (selected), and 'Forum Diskusi'. The main content area displays a thread with two posts. The first post is from 'Affan Ian Amara' (Mahasiswa USK - 10 Merit) asking: 'Apa bedanya Shinny sama R biasa?'. The second post is from 'Syarif Hidayat' (Siswa SMAN 1 Banda Aceh • 2 Hari) responding with: 'Ada yang bisa jelaskan rumus ini?'. Below the posts, there is a mathematical formula for calculating variance ( $S^2$ ) displayed in LaTeX:

$$S^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}$$

At the bottom of the page, there are navigation links for '1', '2', and '3'.

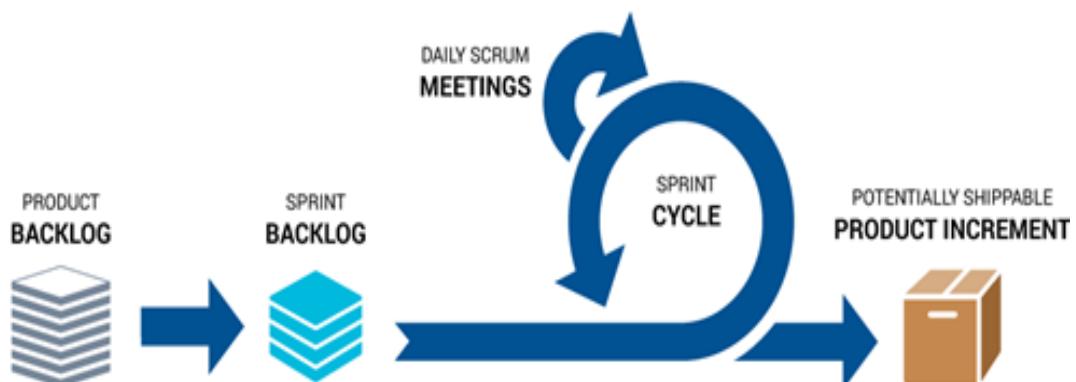
**Gambar 3.6.** Tampilan halaman *Forum Diskusi*



**Gambar 3.7.** Tampilan halaman *Komentar Forum*

### 3.3.5. Pengembangan Sistem

Pada penelitian “Penambahan Fitur Forum Diskusi Pada Sistem Aplikasi Pembelajaran Statistika Rwikistat Berbasis *Website*” peneliti menggunakan metode *SCRUM*. Metode *SCRUM* adalah salah satu pendekatan pengembangan perangkat lunak yang populer dalam manajemen proyek yang diadaptasi dari dunia pengembangan perangkat lunak. Metode ini terkenal karena pendekatannya yang fleksibel, kolaboratif, dan iteratif dalam menghadapi kompleksitas proyek. Scrum menggunakan pendekatan iteratif dan inkremental dalam pengembangan perangkat lunak, yang berarti proyek dipecah menjadi serangkaian iterasi pendek yang disebut “*sprint*”. Setiap *sprint* biasanya berlangsung selama satu hingga empat minggu, dengan tujuan menghasilkan potongan fungsional dari perangkat lunak yang dapat digunakan oleh pengguna. Visualisasi metode *SCRUM* dapat dilihat pada gambar 3.2:



**Gambar 3.8.** Tahapan Metode SCRUM

Tahapan-tahapan dalam metode Scrum meliputi:

**Product Backlog** *Product Backlog* adalah daftar prioritas yang berisi semua kebutuhan, fitur, dan perubahan yang diinginkan untuk produk yang sedang dikembangkan. *Product Owner* bertanggung jawab untuk mengelola *Product Backlog*, memprioritaskan item berdasarkan nilai bisnis, dan menyediakan deskripsi yang jelas untuk setiap item. *Product Backlog* terus diperbarui dan diubah selama siklus pengembangan produk. *Product Backlog* dari aplikasi yang akan dibuat adalah sebagai berikut:

- Tampilan *Landing Page*.
- Tampilan *Login Page*.
- Tampilan halaman *Modul Belajar*.
- Tampilan halaman *R Compiler*.
- Tampilan halaman *Forum Diskusi*.
- Tampilan halaman *Komentar Forum*

**Sprint Backlog** *Sprint Backlog* adalah subset dari *Product Backlog* yang dipilih untuk dikerjakan selama *sprint* tertentu. Tim *Scrum* berkolaborasi untuk memilih dan memperkirakan item dari *Product Backlog* yang akan menjadi bagian dari *Sprint Backlog*. *Sprint Backlog* berisi tugas-tugas yang harus diselesaikan untuk menghasilkan potongan fungsional yang dapat digunakan dari produk. *Sprint Backlog* membantu tim untuk fokus dan mengelola pekerjaan selama *sprint*.

**Sprint Cycle** *Sprint Cycle* adalah periode waktu yang ditetapkan untuk menyelesaikan sebuah *sprint*. Durasi *sprint* biasanya berkisar antara satu hingga empat minggu, dengan durasi yang konsisten pada setiap siklus. Selama *sprint*, tim *Scrum* bekerja untuk menyelesaikan tugas-tugas yang ada dalam *Sprint Backlog* dengan tujuan menghasilkan *Potentially Shippable Product Increment* (Inkrement Produk yang Potensial untuk Dikirimkan) pada akhir *sprint*.

**Sprint Review** *Sprint Review* adalah pertemuan yang diadakan pada akhir setiap *sprint*. Tim *Scrum* memperlihatkan hasil kerja mereka kepada pemangku kepentingan, termasuk potongan fungsional yang telah selesai dari produk. Pemangku kepentingan memberikan umpan balik dan evaluasi terhadap pekerjaan yang telah dilakukan. *Sprint Review* membantu memvalidasi dan mengarahkan pengembangan produk berikutnya.

**Potentially Shippable Product Increment** *Potentially Shippable Product Increment* adalah hasil dari pengembangan selama satu *sprint* yang siap untuk dikirimkan ke pengguna atau pelanggan. Ini berarti potongan fungsional perangkat lunak yang dikembangkan dalam *sprint* tersebut sudah memenuhi kriteria yang diperlukan untuk diuji dan digunakan. Setiap *sprint* bertujuan untuk menghasilkan *Potentially Shippable Product Increment* yang dapat ditinjau oleh pemangku kepentingan.

### 3.3.6. Pengujian Fungsionalitas

Pentingnya pengujian dalam pengembangan perangkat lunak dapat diukur berdasarkan beberapa prinsip utama. Salah satunya adalah pengujian fungsionalitas yang dilakukan menggunakan metode *black box testing*. Tujuan dari pengujian ini adalah untuk memastikan bahwa sistem aplikasi berjalan dan beroperasi sesuai dengan yang diharapkan dan disepakati, tanpa perlu melihat struktur internal seperti kode, manajemen basis data, atau hal-hal yang terkait dengan *source code* aplikasi.

### 3.3.7. Pengujian Usability

Pengujian *Usability* adalah metode yang digunakan untuk mengevaluasi produk atau jasa dengan menguji calon pengguna. Dalam pengujian ini, digunakan metode *Usability Metric for User Experience (UMUX)* yang terdiri dari empat pertanyaan dengan skala *Likert* dari 1 hingga 7. Data dikumpulkan dari pengguna yang menggunakan sistem yang diuji, dengan jumlah responden berkisar antara 10 hingga 2420 orang. Pengujian dapat dilakukan secara *online* atau *offline* menggunakan kuesioner melalui *Google Form*.

### 3.3.8. Analisis Usability

Setelah mengumpulkan data dari responden, langkah selanjutnya adalah mengolah data tersebut dengan menggunakan rumus tertentu untuk mendapatkan skor atau penilaian mengenai kebergunaan sistem aplikasi yang dikembangkan. Berikut adalah ketentuan dalam menghitung skor atau penilaian *UMUX*:

- Setiap item dengan nomor ganjil dihitung dengan rumus [skor pengguna - 1], sedangkan item dengan nomor genap dihitung dengan rumus [7 - skor pengguna].
- Setiap skor penilaian pengguna untuk setiap item dijumlahkan, dan hasilnya dibagi dengan 24 (skor tertinggi).
- Hasil bagi tersebut kemudian dikalikan dengan 100.

- Selanjutnya, hasil tersebut dirata-ratakan untuk semua pengguna yang terlibat dalam pengujian.
- Hasil skor UMUX kemudian dinilai baik atau buruk dengan menggunakan skala 0-100 seperti yang umum digunakan.

### 3.3.9. Penetraion testing

*Penetration testing* merupakan salah satu pengujian yang sangat penting sebelum sebuah *website* di-deploy. Dalam konteks ini, aplikasi *Acunetix* akan digunakan sebagai alat utama untuk melakukan pengujian tersebut. Tujuan dari pengujian ini adalah untuk mengidentifikasi dan mengevaluasi kerentanan keamanan yang mungkin ada dalam *website* tersebut. Dengan melakukan *penetration testing* menggunakan *Acunetix*, tim pengembang dapat menguji kekuatan keamanan *website*, mendeteksi kerentanan seperti *SQL injection*, *cross-site scripting* (XSS), serta mengidentifikasi area-area yang rentan terhadap serangan. Hal ini membantu meningkatkan keamanan *website* sebelum diperkenalkan ke lingkungan produksi, sehingga meminimalkan risiko terhadap serangan *cyber* yang berpotensi merugikan.

## **BAB IV**

## **HASIL DAN PEMBAHASAN**

### **4.1. ANALISIS KEBUTUHAN**

Berdasarkan analisis kebutuhan yang dilakukan, hasil yang didapatkan akan digunakan sebagai dasar untuk merancang aplikasi yang akan dibangun. Berikut adalah hasil dari analisis kebutuhan yang telah dilakukan:

#### **4.1.1. Kelompok *user***

Kelompok *user* pada aplikasi Rwikistat yang telah diidentifikasi sebelumnya, terdapat 2 kelompok *user* yang menggunakan aplikasi ini:

##### **1. Mahasiswa / Dosen**

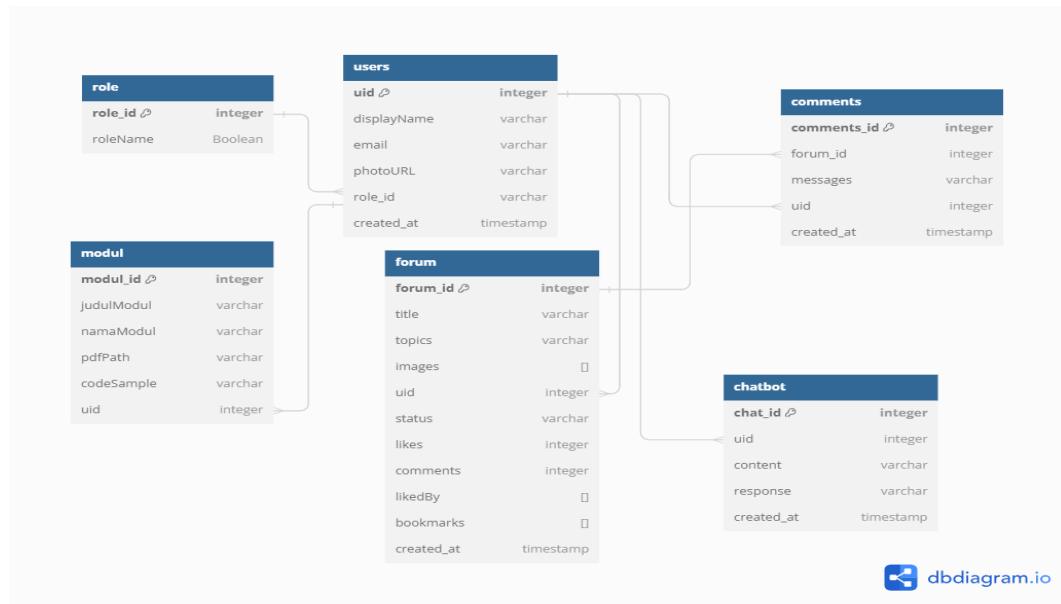
Kelompok *user* ini, memiliki akses hampir ke semua fitur aplikasi Rwikistat. *User* ini bisa melakukan login menggunakan NIM atau akun google, *user* ini juga bisa menggunakan fitur-fitur seperti *R compiler*, membaca modul, bertanya di forum diskusi, menggunakan *chatbot*, dan melihat halaman profile sendiri maupun profile user lain.

##### **2. Admin**

Admin bisa mengakses semua fitur yang ada di aplikasi Rwikistat, yang membedakan kelompok *user* admin dengan kelompok *user* mahasiswa dan dosen adalah admin bisa menambahkan dan menghapus modul, dan admin juga bisa memverifikasi user.

#### **4.1.2. *Entity Relationship Diagram***

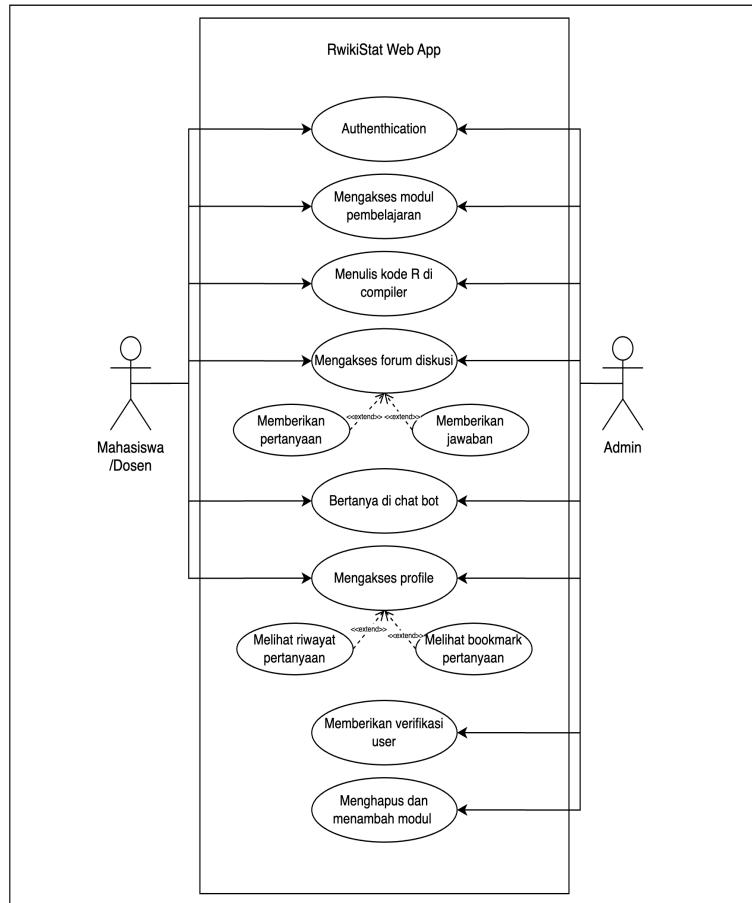
*Entity Relationship Diagram* (ERD) adalah sebuah diagram struktural yang digunakan untuk merancang sebuah database. ERD digunakan untuk mendeskripsikan data yang akan disimpan dalam sebuah sistem maupun batasannya. Komponen utama yang terdapat di dalam sebuah ERD adalah entity set, relationship set, dan juga constraints. Fungsinya adalah untuk memvisualisasikan hubungan antara entitas dalam sebuah database dan menggambarkan bagaimana entitas tersebut saling terkait. ERD juga membantu dalam memodelkan struktur data, mengidentifikasi hubungan antara entitas, dan menentukan atribut-atribut yang dimiliki oleh setiap entitas. ERD juga dapat digunakan untuk mengkomunikasikan struktur data kepada pemangku kepentingan (Latukolan, 2019). Pada gambar 4.1 berikut adalah ERD dari aplikasi Rwikistat yang telah dibangun.



**Gambar 4.1. Entity Relationship Diagram**

#### 4.1.3. Use Case Diagram

*Use case* diagram digunakan untuk memvisualisasikan interaksi antara *user* dengan sistem perangkat lunak. Diagram ini menggambarkan fungsionalitas sistem dari perspektif *user*, menunjukkan bagaimana *user* akan berinteraksi dengan sistem untuk mencapai tujuan tertentu. *Use case* diagram membantu dalam memahami kebutuhan *user*, mengidentifikasi fitur-fitur utama sistem, dan merencanakan pengembangan perangkat lunak berdasarkan interaksi *user*. Diagram ini juga dapat digunakan untuk berkomunikasi dengan pemangku kepentingan dan tim pengembang mengenai fungsionalitas sistem secara visual dan mudah dipahami (Prasad, 2020). *Use case* diagram dari aplikasi Rwikistat yang telah dibangun dapat dilihat pada gambar 4.2.



**Gambar 4.2. Use Case Diagram**

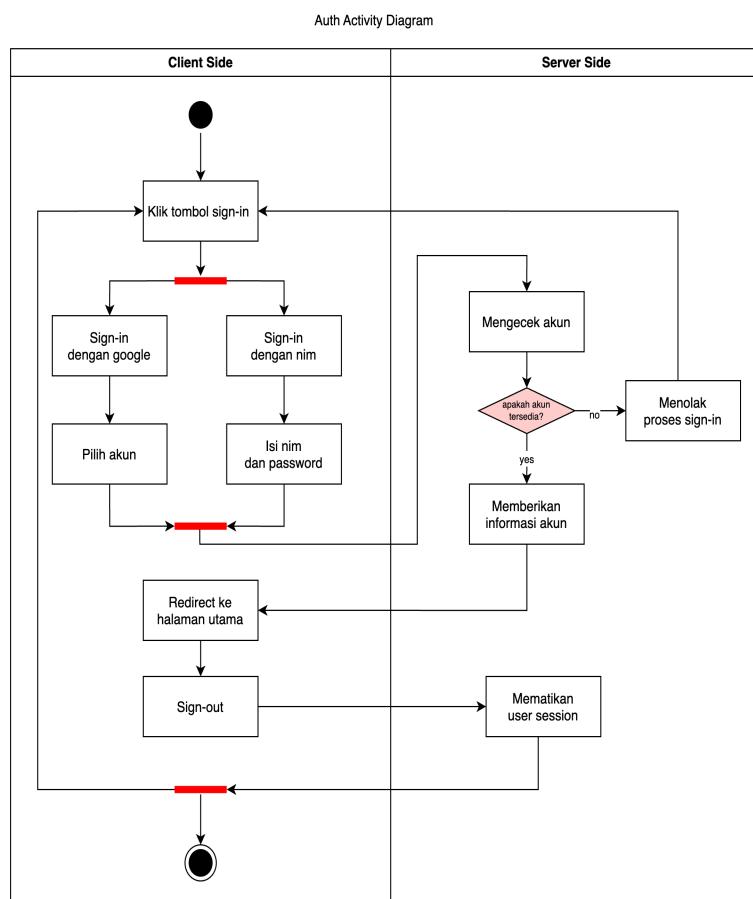
Pada gambar 4.2 menjelaskan aktivitas-aktivitas yang dapat dilakukan oleh mahasiswa, dosen, dan admin saat menjalankan aplikasi Rwikistat. Semua kelompok *user* bisa melakukan login, akan tetapi yang membedakan admin dan kelompok *user* lainnya adalah admin hanya bisa login menggunakan akun google, sedangkan kelompok *user* lainnya bisa login menggunakan NIM. Setelah login, *user* bisa menggunakan fitur-fitur seperti *R compiler*, membaca modul, bertanya dan memberikan komentar di forum diskusi, menggunakan *chatbot*, dan mengakses halaman *profile*. Di halaman *profile* *user* bisa melihat riwayat postingan dan bookmark. Hal yang membedakan antara admin dengan kelompok *user* lainnya adalah admin bisa menambahkan dan menghapus modul, dan admin juga bisa memverifikasi user.

#### 4.1.4. Activity Diagram

*Activity diagram* adalah salah satu jenis diagram yang digunakan dalam *Unified Modeling Language* (UML) untuk memodelkan alur kerja atau aliran aktivitas dalam suatu proses. *Activity diagram* digunakan untuk menggambarkan urutan aktivitas, keputusan, dan aliran kontrol dalam suatu proses atau sistem.

Fungsinya adalah untuk memvisualisasikan proses bisnis atau alur kerja sistem secara jelas dan terstruktur, sehingga memudahkan pemahaman dan analisis terhadap aktivitas yang terjadi. Dalam pengembangan perangkat lunak, *Activity diagram* juga dapat digunakan sebagai dasar untuk menghasilkan *test case* dan memvalidasi proses bisnis atau alur kerja yang diimplementasikan dalam perangkat lunak (Ahmad, 2019). Berikut adalah Activity Diagram yang menginterpretasikan aktivitas-aktivitas yang terjadi pada aplikasi Rwikistat.

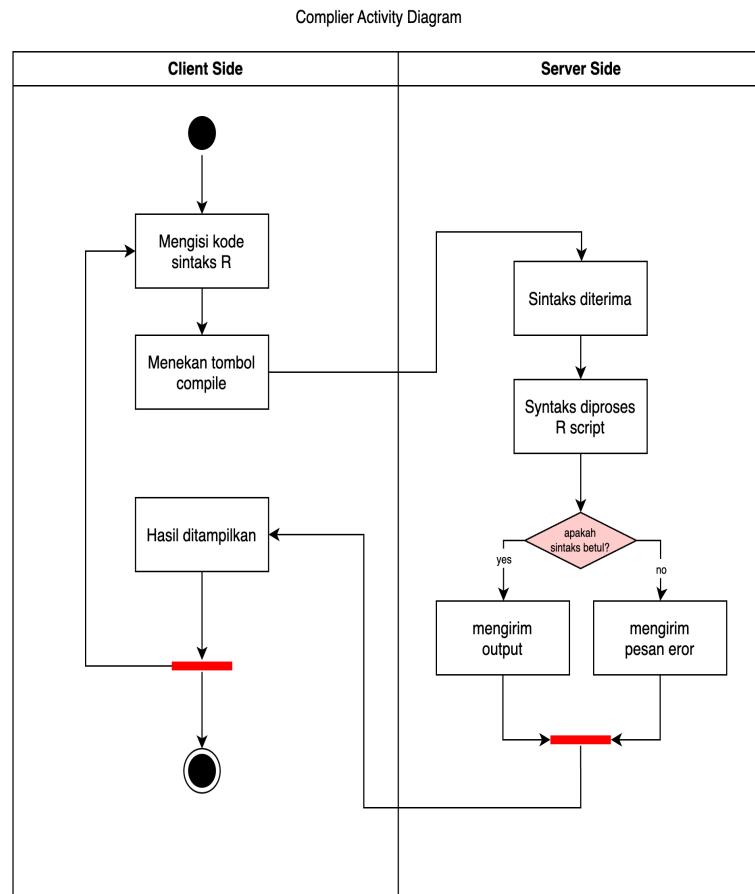
### 1. *Activity Diagram Login*



**Gambar 4.3. *Activity Diagram Login***

*Activity diagram* pada gambar 4.3 menjelaskan aktivitas *user* saat melakukan kegiatan login ke website Rwikistat. Langkah pertama yang dilakukan *user* adalah memilih metode login, apakah menggunakan NIM atau akun google. Jika *user* memilih login menggunakan NIM, maka *user* bisa langsung mengisi form NIM dan Password. Jika *user* memilih login menggunakan akun google, maka *user* bisa menekan tombol login dengan google, setelah itu *user* akan diperintahkan untuk memilih akun google yang akan digunakan, jika proses login sukses akan diarahkan ke halaman *compiler*.

## 2. Activity Diagram R Compiler

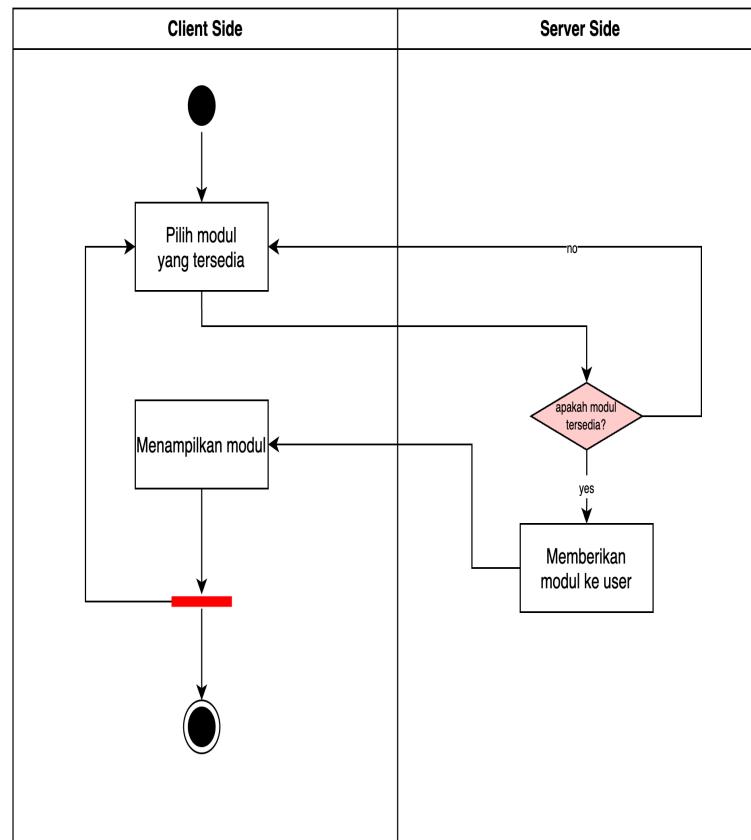


**Gambar 4.4. Activity Diagram R Compiler**

Activity diagram pada gambar 4.4 menjelaskan aktivitas user saat menjalankan fitur r compiler. Hal yang pertama dilakukan saat akan menjalankan fitur ini adalah, user diharuskan mengisi syntax R, setelah syntax di isi, user bisa menekan tombol run untuk menjalankan syntax R yang telah diisi. Setelah terkirim,syntax R akan dikirim dan di proses di API, setelah hasilnya didapatkan, output akan dikirimkan kembali ke website.Jika syntax R yang diisi tidak ada kesalahan, maka user akan mendapatkan hasil dari syntax R yang telah diisi. Jika syntax R yang diisi terdapat kesalahan, maka user akan mendapatkan pesan kesalahan dari syntax R yang telah diisi.

## 3. Activity Diagram Pilih Modul Pembelajaran

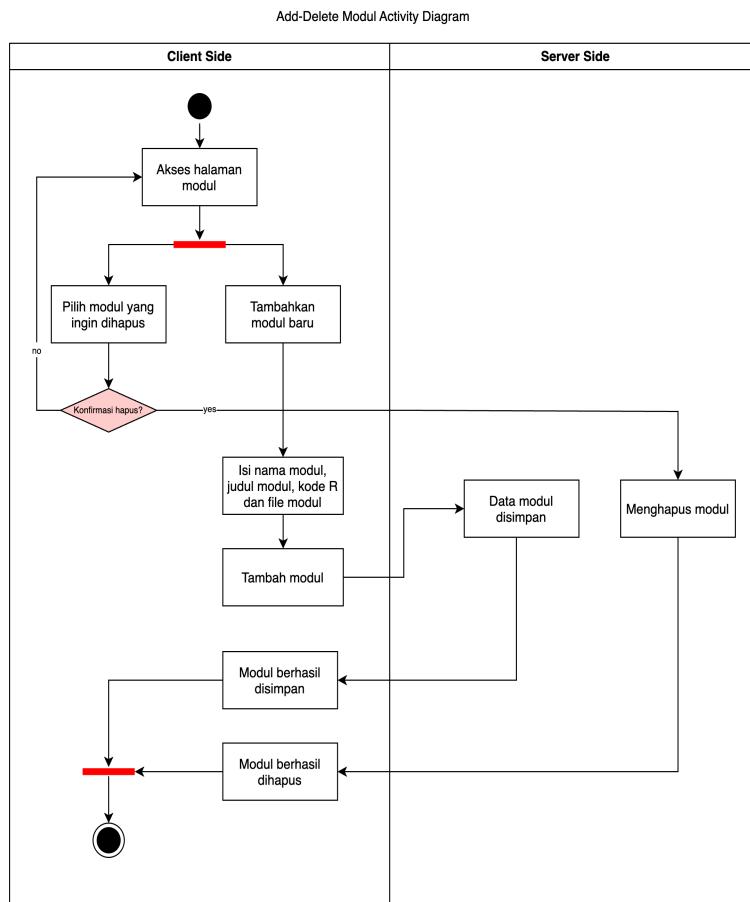
Select Modul Activity Diagram



**Gambar 4.5.** Activity Diagram Pilih Modul

*Activity diagram* pada gambar 4.5 menjelaskan aktivitas *user* saat memilih modul pembelajaran yang akan dibaca. Sebelum mulai membaca, *user* diharuskan memilih modul dahulu, setelah *user* memilih modul, *user* akan diarahkan ke halaman modul. Di halaman modul, *user* bisa melihat isi dari modul yang telah dipilih.

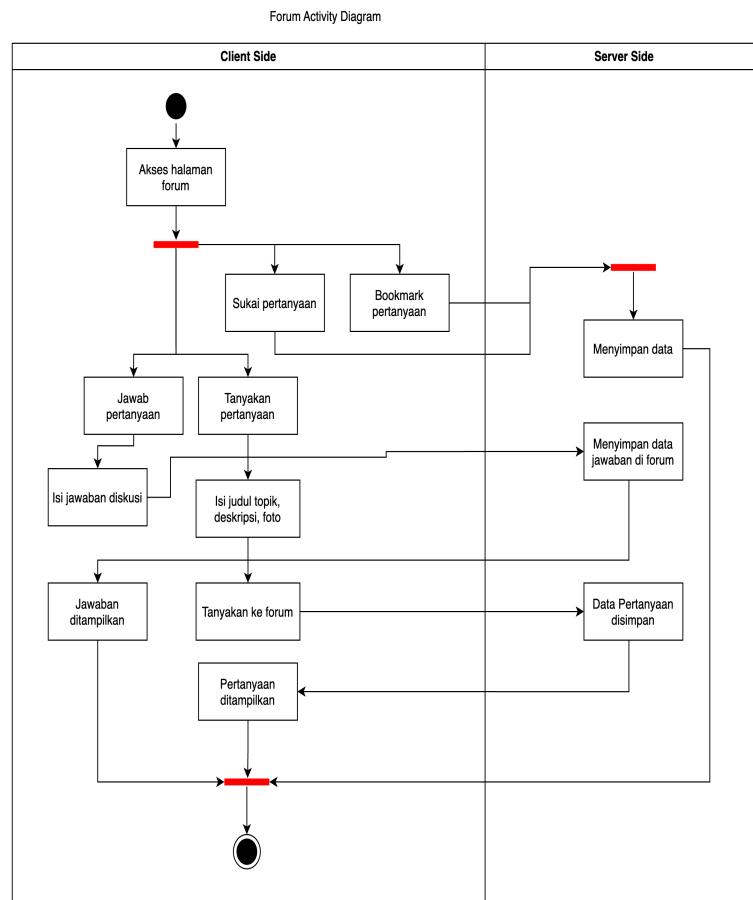
#### 4. *Activity Diagram* tambah dan hapus modul



**Gambar 4.6.** *Activity Diagram tambah dan hapus modull*

*Activity diagram* pada gambar 4.6 menjelaskan aktivitas yang dilakukan *user*, aktivitas ini hanya dilakukan oleh admin. Admin bisa menambahkan modul baru, dan admin juga bisa menghapus modul yang telah ada. Jika admin ingin menambahkan modul baru, admin bisa menekan tombol tambah modul dan mengisi form yang telah disediakan. Pada form pengisian modul, admin diharuskan menginput modul dengan ekstensi pdf, setelah *form* di isi data modul akan dikirim ke server dan modul akan otomatis tersimpan di database. Jika admin ingin menghapus modul, admin bisa menekan tombol hapus di modul yang dipilih, admin bisa menekan tombol hapus, setelah itu modul akan dihapus dari database.

## 5. *Activity Diagram* Forum Diskusi

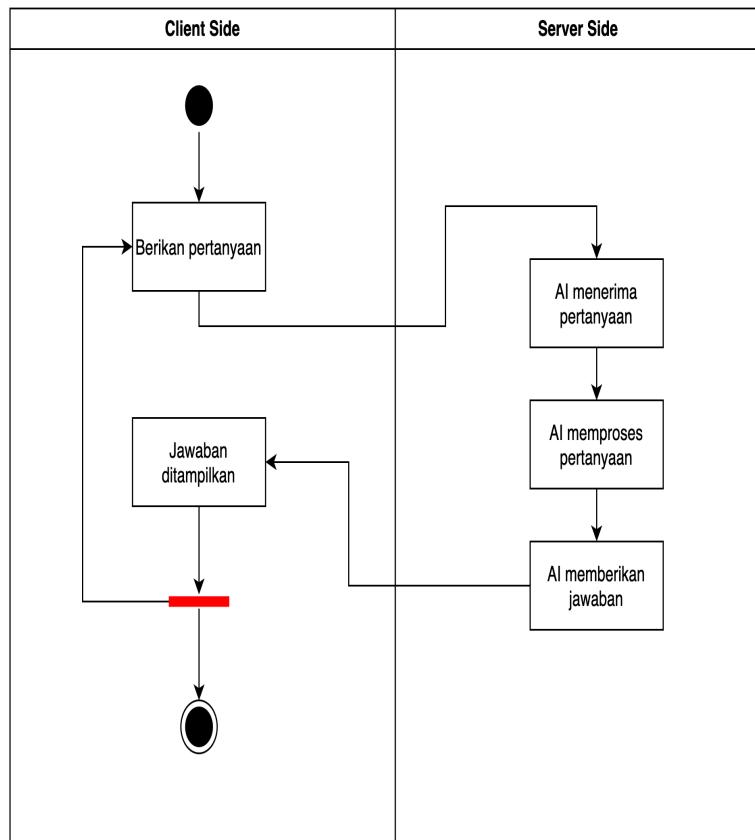


**Gambar 4.7.** *Activity Diagram* forum diskusi

Pada gambar 4.7 menjelaskan *activity diagram user* saat di halaman forum diskusi. Di halaman forum diskusi, *user* bisa melakukan beberapa aktivitas, diantaranya memberikan pertanyaan, menjawab pertanyaan, memberikan *like* terhadap satu pertanyaan, dan *bookmark* pertanyaan. Jika *user* ingin memberikan pertanyaan, *user* bisa mengklik kolom tambah pertanyaan dan mengisi *form* yang tersedia, *user* bisa mengisi judul, topik pembahasan dan gambar pendukung jika diperlukan. Jika *user* ingin memberikan jawaban atau komentar terhadap satu pertanyaan, *user* bisa memilih salah satu pertanyaan yang akan dijawab dan menginput komentar yang diinginkan. Selanjutnya jika ingin memberikan *like* dan *bookmark* terhadap satu pertanyaan, *user* bisa menekan tombol *like* dan *bookmark* yang tersedia di setiap pertanyaan.

## 6. *Activity Diagram* Chatbot

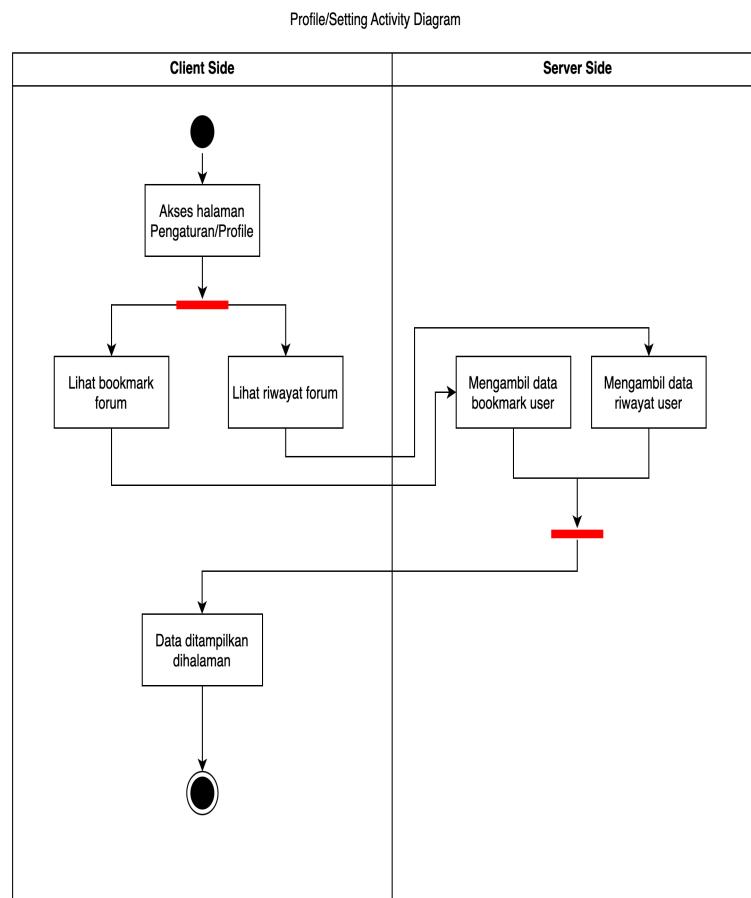
Chat Bot Activity Diagram



**Gambar 4.8.** *Activity Diagram* chatbot

Pada gambar 4.8 menjelaskan *activity diagram user* saat menggunakan fitur chatbot. *User* bisa mengajukan pertanyaan ke *chatbot* dengan mengetikkan pertanyaan yang diinginkan, setelah pertanyaan diterima oleh *AI*, pertanyaan akan diproses pertanyaan yang diberikan *user* dan memberikan jawaban yang sesuai dengan pertanyaan yang diberikan *user*. Jawaban yang diberikan akan ditampilkan di halaman yang sama saat memberikan pertanyaan ke *chatbot*.

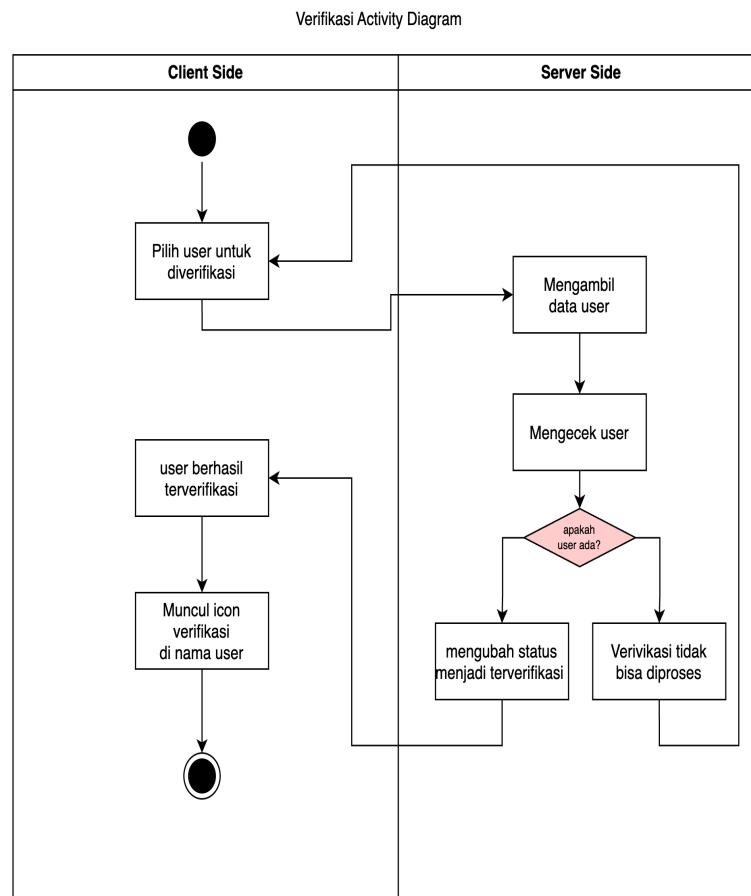
#### 7. *Activity Diagram Profile*



**Gambar 4.9. Activity Diagram profile**

Gambar 4.9 menjelaskan *activity diagram user* saat mengakses halaman *profile*. Pada halaman *profile*, *user* bisa melihat pertanyaan forum diskusi yang di posting dan di *bookamrk user* tersebut. Untuk melihat pertanyaan yang diberikan *user* bisa menekan tombol *posted*, dan untuk melihat pertanyaan yang di *bookmark user* bisa menekan tombol *bookmark*. Setiap pilihan yang dipilih akan menampilkan data yang sesuai dengan yang diinginkan, baik pertanyaan yang di posting maupun pertanyaan yang di *bookmark*.

#### 8. Activity Diagram Verifikasi User



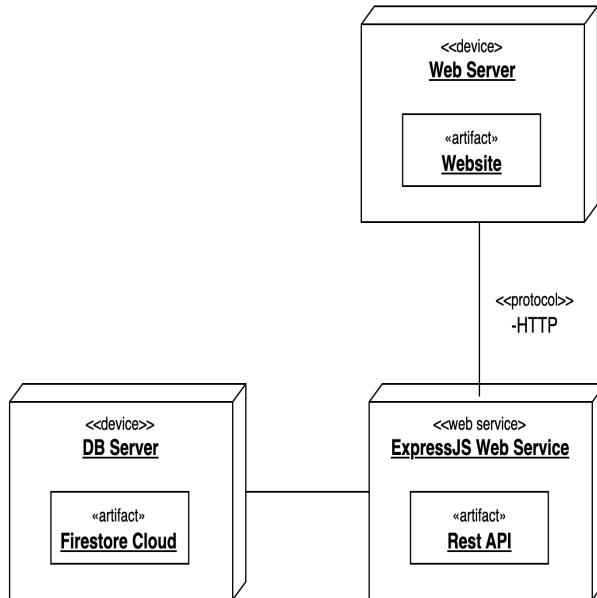
**Gambar 4.10.** *Activity Diagram verifikasi user*

*Activity Diagram* pada gambar 4.10 menunjukkan aktivitas *user* saat melakukan verifikasi. Verifikasi dilakukan oleh admin, untuk melakukan verifikasi admin harus memilih *user* yang akan diverifikasi terlebih dahulu, *user* yang dipilih akan dikirim ke server dan akan diperiksa apakah user tersedia atau tidak. Jika tersedia, server akan mengembalikan respon sukses dan akan muncul *icon* centang di halaman *profile user* yang telah diverifikasi. Jika *user* tidak tersedia, server akan mengembalikan respon gagal.

#### 4.1.5. Deployment Diagram

*Deployment diagram* digunakan untuk memodelkan struktur fisik dari sistem perangkat lunak, termasuk komponen fisik seperti server, perangkat keras, dan koneksi fisik seperti kabel dan jaringan nirkabel . Diagram ini juga digunakan untuk memetakan antara komponen fungsional dengan komponen fisik dari sistem perangkat lunak . *Deployment diagram* membantu dalam memahami bagaimana sistem perangkat lunak diimplementasikan secara fisik dan bagaimana komponen-komponen tersebut berinteraksi satu sama lain. Diagram ini juga dapat digunakan untuk berkomunikasi dengan pemangku kepentingan dan tim

pengembang mengenai struktur fisik sistem secara visual dan mudah dipahami (Ozkaya, 2020). *Deployment* diagram dari aplikasi Rwikistat yang telah dibangun dapat dilihat pada gambar 4.11.



**Gambar 4.11. Deployment Diagram**

## 4.2. PEMBUATAN APLIKASI

Pembuatan aplikasi yang dilakukan peneliti ialah berupa *back end* dan *front end* aplikasi. *Back end* aplikasi yang dibuat berupa REST API yang dibuat menggunakan *framework* NodeJs yaitu ExpressJS dan database Firebase. Sedangkan *front end* aplikasi dibangun menggunakan *framework* NextJS dengan mengimplementasikan konsep *Static Site Generation*.

### 4.2.1. Pembuatan *back end* aplikasi

Hasil pembuatan *back end* aplikasi menghasilkan REST API yang terkoneksi dengan database Firebase. REST API yang dihasilkan ini digunakan untuk menjalankan perintah *request* dari sisi *client* baik website maupun aplikasi mobile dengan mengembalikan respon data dalam format JSON. Akses REST API yang dihasilkan dibatasi dengan menggunakan *Firebase ID Token*, sehingga hanya pengguna yang memiliki token tersebut yang dapat mengakses *endpoint - endpoint* REST API yang dihasilkan.

REST API yang telah dibuat terdiri dari 31 *endpoint*, setiap *endpoint* memiliki fungsi yang berbeda-beda. Berikut adalah daftar *endpoint* yang telah dibuat:

**Tabel 4.1. REST API**

Endpoint	Token	Method	Parameter	Fungsi
/modul	Ya	POST	codeSampel, Judul, pdfFile, namaModul	Menginput data modul baru
/modul	Ya	GET	-	Mendapatkan semua data modul
/modul/:id	Ya	GET	id	Mendapatkan data dengan id tertentu
/modul/:id	Ya	DELETE	id	Menghapus data dengan id tertentu
/compiler	Ya	POST	code	Menginput code R
/forum	Ya	POST	topics,title,userId,images	Memberikan pertanyaan baru
/forum/page	Ya	GET	page	Mengambil semua data forum
/forum/search	Ya	POST	query	Mencari data forum tertentu
/forum/user/:userId	Ya	GET	userId	Mengambil satu data user yang memposting pertanyaan di forum
/forum/:id	Ya	GET	id	Mengambil satu data forum berdasarkan id
/forum/:id/comments	Ya	POST	id,text,userId	Menambahkan komentar ke salah satu postingan forum
/forum/:id/comments	Ya	GET	id	Menampilkan komentar salah satu postingan
/forum/like/:id	Ya	POST	id,userId	Memberikan <i>like</i> ke satu postingan forum
/forum/unlike/:id	Ya	POST	id,userId	Membatalkan <i>like</i> ke satu postingan forum
/forum/like/:id/is-liked	Ya	GET	id,userId	Memeriksa apakah <i>user</i> sudah memberikan <i>like</i>
/forum/bookmark/:id	Ya	POST	id,userId	Memberikan <i>bookmark</i> ke satu postingan forum
/forum/unbookmark/:id	Ya	POST	id,userId	Membatalkan <i>bookmark</i> ke satu postingan forum

/forum/:id/is-bookmarked	Ya	GET	id,userId	Memeriksa apakah user sudah memberikan bookmark
/forum/bookmarks/:userId	Ya	GET	userId,page	Mengambil data forum yang di bookmark oleh user
/forum/posted/:userId	Ya	GET	userId,page	Mengambil data forum diposting oleh user
/user/:userId	Ya	POST	userId	Memverifikasi user
/user	Ya	GET	-	Mengambil data semua user
/user/:id	Ya	GET	userId	Mengambil data satu user
/user/posted/:userId	Ya	GET	userId	Mengambil data postingan user
/user/:uid/score	Ya	GET	userUd	Menampilkan score yang dimiliki user
/chatbot	Ya	POST	messages	Memberikan pertanyaan ke chatbot
/chatbot/:userId	Ya	GET	userId	Mengambil jawaban yang diberikan oleh chatbot
/history	Ya	POST	userId	Menyimpan gambar hasil kompilasi
/history	Ya	GET	userId	Menampilkan gambar yang disimpan user
/history/delete/:id	Ya	DELETE	id	Menghapus salah satu gambar
/history/download/:id	Ya	GET	id	Men-download gambar

#### 4.2.2. Pembuatan *front end* aplikasi

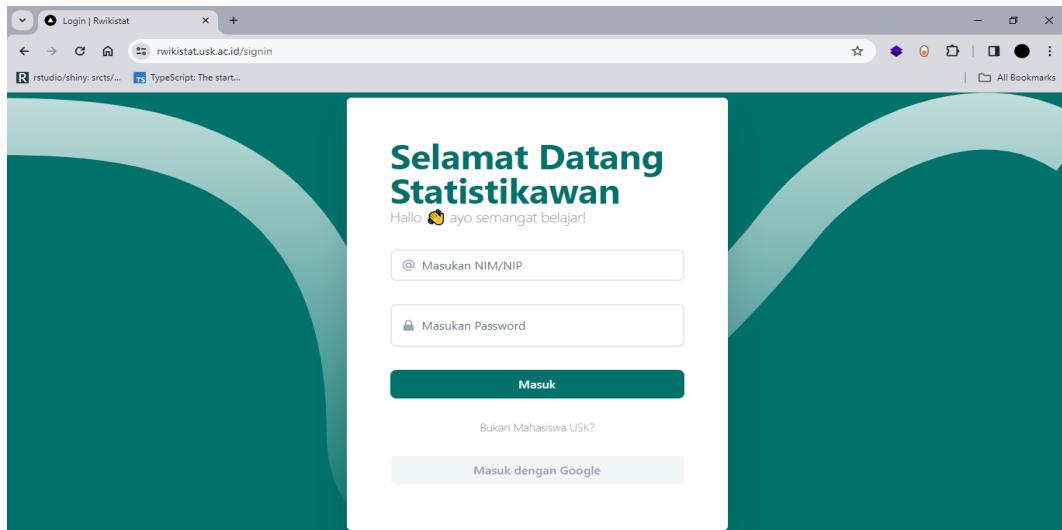
Pembuatan *front end* aplikasi dilakukan dengan menggunakan *framework* Next.js dan mengusung konsep *Static Site Generation*. *Front end* yang dihasilkan mencakup beberapa halaman *website* yang dirancang untuk memberikan bantuan dan kemudahan kepada *user* dalam menggunakan aplikasi. *Website* Rwikistat bisa diakses memalui url "<https://rwikistat.usk.ac.id>", halaman awal yang ditampilkan saat pertama kali membuka *website* Rwikistat, sebagaimana terlihat pada Gambar 4.12, menampilkan informasi tentang aplikasi Rwikistat. Di samping itu, halaman

ini juga menyajikan sejumlah fitur yang dapat diakses oleh *user*. Fitur-fitur tersebut dirancang untuk memberikan gambaran komprehensif tentang aplikasi, memungkinkan *user* untuk dengan mudah mengakses dan memanfaatkan berbagai fungsionalitas yang disediakan.



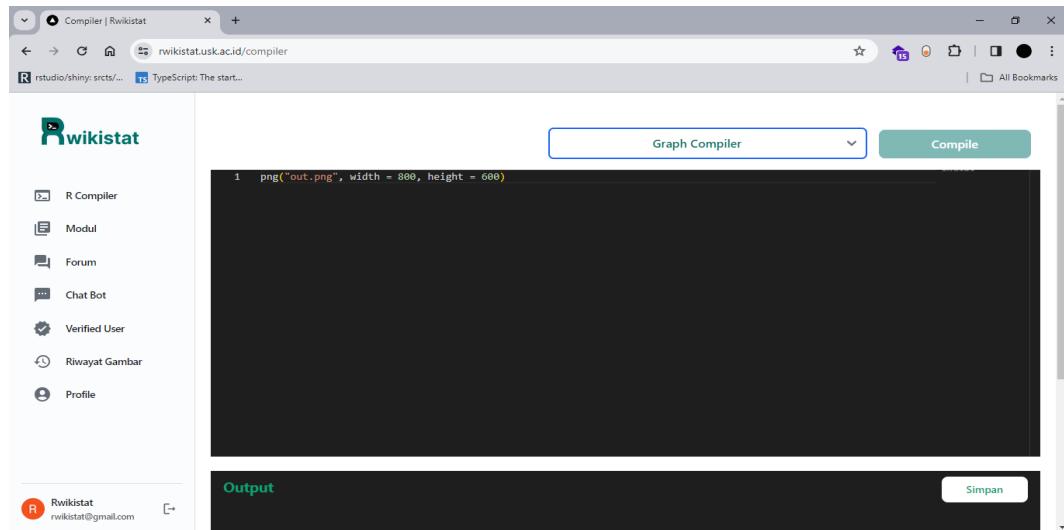
**Gambar 4.12.** Halaman *dashboard*

Gambar 4.13 menampilkan halaman *login* yang berfungsi sebagai pintu bagi *user* untuk masuk ke dalam aplikasi Rwikistat dan memanfaatkan berbagai fitur yang disediakan. Halaman *login* ini dirancang dengan sebuah *form* yang memungkinkan *user*, baik mahasiswa maupun dosen Universitas Syiah Kuala (USK), untuk melakukan *login* menggunakan NIM/NIP dan password yang sesuai dengan yang digunakan untuk mengakses KRS. Selain itu, *user* juga memiliki opsi untuk melakukan *login* melalui akun Google dengan menggunakan tombol yang tersedia. Dengan demikian, halaman *login* ini menyediakan dua cara alternatif bagi *user* untuk mengakses aplikasi, sesuai dengan kategori *user* yang dapat memanfaatkan layanan Rwikistat.



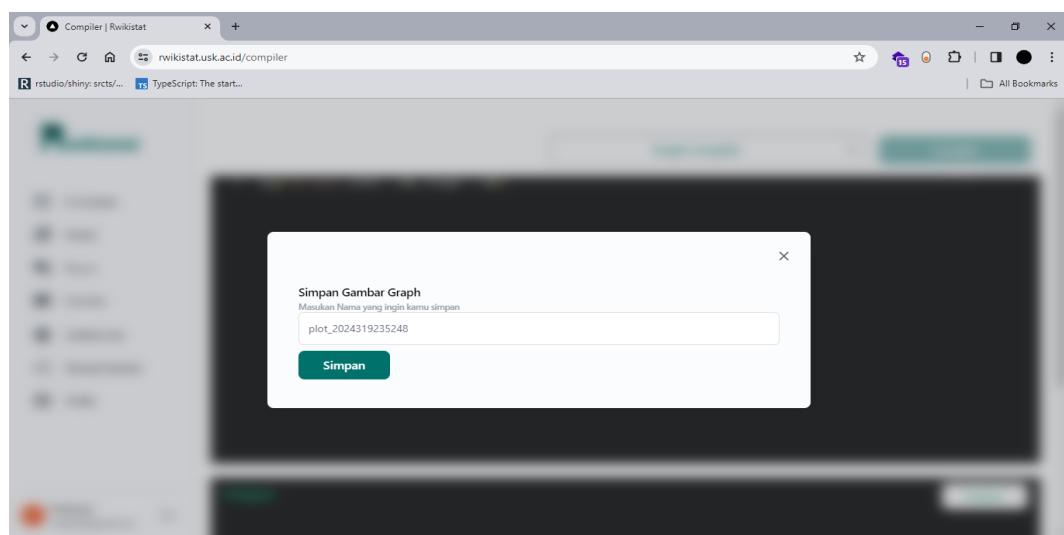
**Gambar 4.13.** Halaman *login*

Setelah melakukan *login* ke dalam aplikasi menggunakan NIM maupun akun google, *user* akan diarahkan ke halaman *compiler* yang ditampilkan pada gambar 4.14. Pada halaman *compiler* tersedia sebuah *form* yang memungkinkan *user* untuk menuliskan *syntax R* yang ingin dijalankan. Di halaman ini, *user* bisa memilih jenis *compiler* yang ingin dijalankan, String *compiler* atau Graph *compiler*. Setelah *user* menuliskan *syntax R* yang diinginkan, *user* dapat menekan tombol *compile* untuk menjalankan *syntax R* yang telah dituliskan. Setelah *syntax R* dikirim ke server, server akan memproses *syntax R* yang telah dikirimkan dan mengembalikan hasilnya ke *website*. Jika *syntax R* yang dikirimkan tidak ada kesalahan, maka *user* akan mendapatkan hasil dari *syntax R* yang telah dikirimkan dan ditampilkan di bagian *output*. Jika *syntax R* yang dikirimkan terdapat kesalahan, maka *user* akan mendapatkan pesan kesalahan dari *syntax R* yang telah dikirimkan dan ditampilkan di bagian *output* juga. Jika *user* memilih Graph *compiler*, akan muncul sebuah tombol yang berfungsi untuk menyimpan gambar yang dihasilkan, gambar yang disimpan bisa dilihat di halaman Riwayat Gambar.



**Gambar 4.14.** Halaman *R compiler*

Jika *user* memilih *graph compiler* dan menekan tombol simpan, akan muncul sebuah menu *popup* yang berfungsi untuk melakukan penyimpanan gambar, sebelumnya menyimpan gambar, *user* bisa memasukkan nama gambar yang disimpan, jika tidak diisi, nama gambar akan disimpan secara *default* dan akan disimpan dengan nama "plot + waktu penyimpanan". Menu simpan gambar dapat dilihat pada gambar 4.15

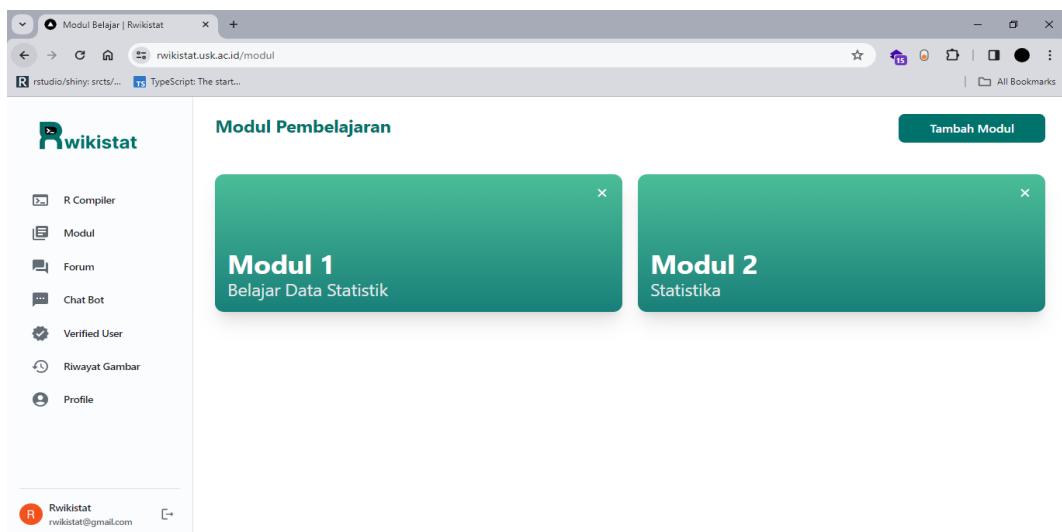


**Gambar 4.15.** Halaman *simpan gambar*

Gambar 4.16 menampilkan halaman modul yang berperan sebagai wadah untuk memilih modul mana yang ingin diakses. Pada halaman ini, *user* memiliki kemampuan untuk menelusuri daftar modul yang telah disediakan dan memilih modul yang diminati. Caranya adalah dengan menekan salah satu modul yang diinginkan, dan setelah itu, *user* akan diarahkan ke halaman detail modul yang

bersangkutan. Untuk *user* yang memiliki peran sebagai admin, tersedia opsi tambahan untuk menambahkan modul baru atau menghapus modul yang sudah ada.

Pada *level* admin, jika ingin menambahkan modul baru, admin dapat menggunakan tombol tambah modul yang akan mengarahkannya ke halaman khusus untuk menambahkan modul. Selain itu, admin juga memiliki kemampuan untuk menghapus modul dengan menekan tombol silang yang tersedia di setiap modul pada halaman modul utama. Dengan adanya fungsionalitas ini, admin dapat dengan mudah mengelola koleksi modul, menjadikan halaman modul sebagai pusat kontrol yang efisien untuk pengelolaan konten.



**Gambar 4.16.** Halaman modul

Setelah *user* memilih salah satu modul yang akan dibaca, *user* akan diarahkan ke halaman detail modul seperti yang ditampilkan pada gambar 4.17. Pada halaman detail modul *user* bisa melihat modul pembelajaran yang dipilih, dibagian bawah modul terdapat sebuah *code editor* yang berfungsi seperti layaknya halaman compiler yang ada pada gambar 4.14 sebelumnya. Dan dibagian paling halaman detail modul bawah juga terdapat bagian yang digunakan untuk menampilkan web shiny yang telah dijalankan. Dan di bagian atas terdapat sebuah menu yang berisi daftar modul yang telah disediakan, *user* bisa memilih modul yang ingin dibaca dengan menekan salah satu modul yang ada di menu tersebut.

**BAB II : STATISTIKA**

**2.1 Pengertian Statistika**

Statistika merupakan ilmu yang mempelajari tentang bagaimana cara mengumpulkan, menyajikan, menganalisis, dan menginterpretasi data yang bertujuan untuk memperoleh suatu informasi yang mudah dipahami oleh masyarakat umum. Sedangkan, statistik merupakan hasil yang diperoleh dari analisis data. Statistika berasal dari istilah-istilah bahasa latin modern yaitu "statisticum collegium" yang artinya dewan negara dan bahasa Italia yaitu "statista" yang artinya negarawan atau politikus. Statistik pertama kali digunakan oleh seorang guru besar dari Marlborough University dan Gottingen yang bernama Gottfried Achenwall. Pada tahun 1749, Gottfried Achenwall menggunakan statistik pertama kali dalam bahasa Jerman sebagai nama kegiatan analisis data kenegaraan atau diartikan dengan ilmu tentang negara. Adapun pengertian statistika dari beberapa referensi sebagai berikut:

Statistik inferensial adalah bagian dari ilmu statistika yang memfokuskan pada pembuatan kesimpulan tentang populasi berdasarkan data sampel. Teori probabilitas dan statistik digunakan untuk mengestimasi parameter populasi dan membuat keputusan mengenai karakteristik dari populasi (Berenson et al., 2014)

Statistik inferensial berhubungan dengan membuat generalisasi tentang karakteristik populasi berdasarkan data sampel yang diambil. Ini menggunakan prinsip probabilitas dan metode statistik untuk menaksir parameter populasi dan membuat kesimpulan tentang karakteristik populasi secara umum (Utts, 2015)

**Gambar 4.17.** Halaman detail modul

Halaman tambah modul yang ditampilkan pada gambar 4.18 merupakan halaman khusus yang hanya dapat diakses oleh admin. Pada halaman ini, admin dapat menambahkan modul baru dengan mengisi *form* yang telah disediakan. Pada *form* pengisian modul, admin diharuskan menginput nama modul, sebagai contoh modul 1, modul 2, modul 3 dan seterusnya. Admin juga harus mengisi judul modul yang akan di *input*, selanjutnya admin juga diharuskan mengisi contoh *code* yang merupakan *syntax R* yang bisa digunakan mahasiswa sebagai contoh. Setelah itu admin juga diharuskan memasukkan url shiny yang sudah dijalankan sebelumnya. Terakhir admin harus memasukkan isi modul dengan menggunakan format *markdown*, dibagian ini admin diharapkan membaca dokumentasi *markdown* terlebih dahulu agar memudahkan saat melakukan pengisian konten modul.

**Tambah Modul Pembelajaran**

Masukan Nama Modul  
contoh : Modul 1, Modul 2 dll..

Masukan Judul Modul  
contoh : Belajar Dasar Statistik

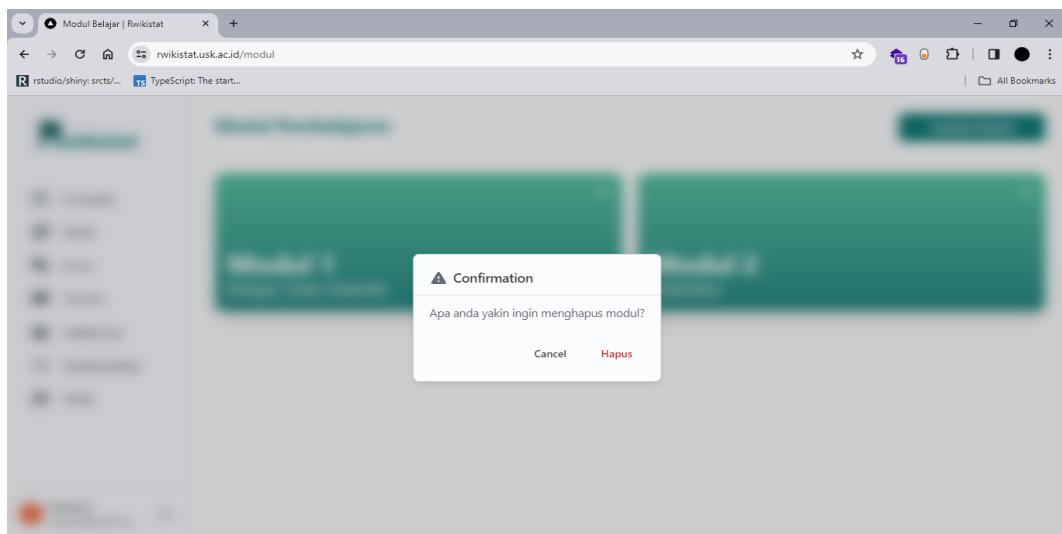
Masukan Url Shiny  
local

Masukan Contoh Kode R  
contoh : 1 + 1

Isi Modul

**Gambar 4.18.** Halaman tambah modul

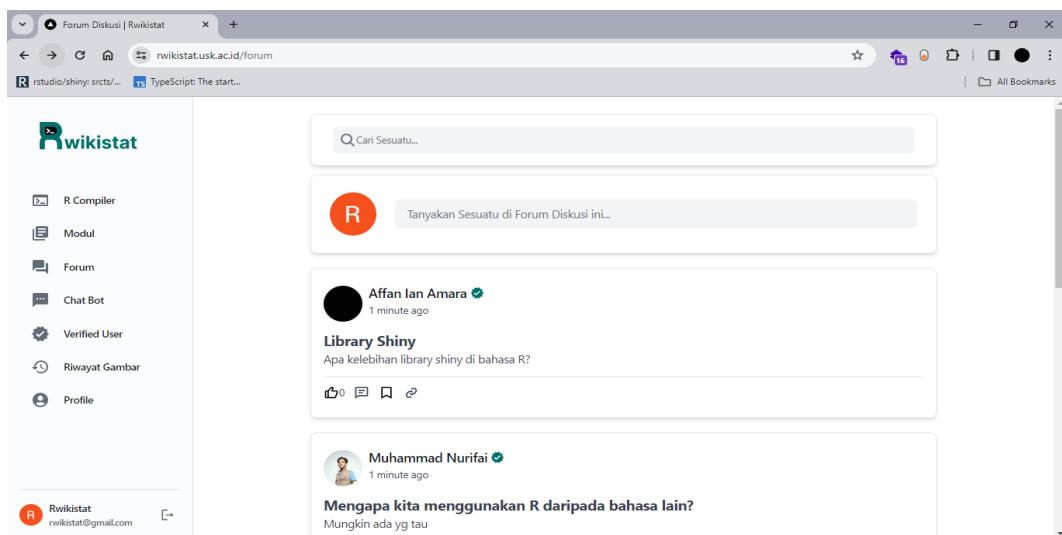
Pada gambar 4.19 menampilkan halaman hapus modul yang berfungi untuk meyakinkan admin apakah akan menghapus modul yang dimaksud. Sama seperti halaman tambah modul, halaman hapus modul juga merupakan halaman khusus yang hanya dapat diakses oleh admin. Untuk menghapus sebuah modul, admin bisa tombol silang yang ada di halaman modul seperti yang ditampilkan di gambar 4.16. Setelah menekan tombol silang, admin akan diarahkan ke halaman hapus modul, di halaman ini admin bisa memilih apakah akan menghapus modul atau tidak. Jika admin ingin menghapus modul, admin bisa menekan tombol hapus, setelah itu modul akan dihapus dari database. Jika admin tidak ingin menghapus modul, admin bisa menekan tombol *cancel*, setelah itu admin akan diarahkan kembali ke halaman modul.



**Gambar 4.19.** Halaman hapus modul

Halaman forum diskusi yang ditampilkan pada gambar 4.20 merupakan halaman yang menampilkan daftar pertanyaan yang tersedia. Di halaman ini *user* bisa melakukan banyak hal, dibagian paling atas terdapat tombol *search* yang bisa digunakan untuk mencari postingan tertentu dan juga bisa mengklik form tambah pertanyaan dan akan diarahkan ke halaman tambah pertanyaan. Lalu dibagian pertanyaan, terdapat beberapa informasi mengenai pertanyaan yang diberikan diantaranya nama *user* yang memberikan, dibawah nama ada waktu kapan pertanyaan diberikan. Disebelah kiri nama *user* juga terdapat foto profil *user*. Selanjutnya untuk pertanyaan yang diberikan ada dua informasi yang diberikan, pertama adalah judul pertanyaan yang ditampilkan dengan tulisan yang lebih tebal dan kedua adalah topik pembahasan. Dibagian *footer* dari pertanyaan terdapat beberapa *icon* yang bisa diklik, yang pertama ada tombol *like* yang digunakan untuk menyukai postingan dan disebelah kanan dari tombol like terdapat informasi berapa jumlah like yang sudah diberikan ke postingan tersebut.

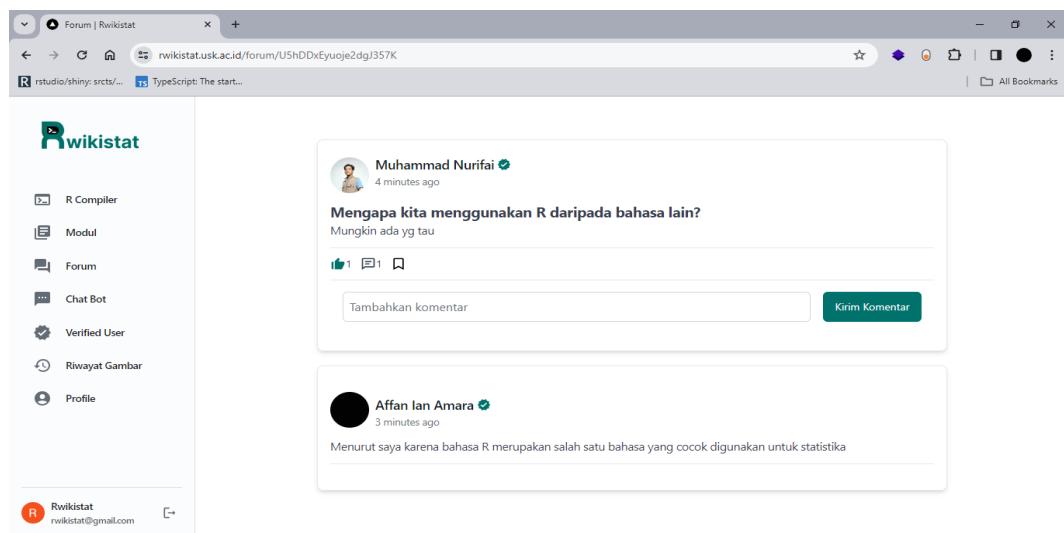
Selanjutnya ada tombol *comment* yang bisa di klik untuk memberikan komentar, jika tombol *comment* di klik, *user* akan diarahkan ke halaman detail dan disebelah tombol *comment* juga terdapat informasi jumlah komentar yang diberikan. Setelah itu ada tombol untuk memberikan *bookmark*, postingan yang di *bookmark* akan tersimpan dan bisa dilihat di halaman *profile* nantinya. Untuk postingan yang sudah diberikan *like* dan *bookmark* akan berubah warna menjadi hitam. Saat tombol *like* dan *bookmark* berwarna hitam, *user* bisa menekan tombol tersebut untuk membatalkan *like* dan *bookmark* yang telah diberikan. Terakhir ada tombol untuk meng-*copy link* dari postingan tersebut, jika tombol ini ditekan maka link dari postingan tersebut akan tersalin dan bisa di *paste* di tempat lain. *Link* yang tersalin bisa digunakan untuk melihat halaman detail dari postingan yang bersangkutan.



**Gambar 4.20.** Halaman forum diskusi

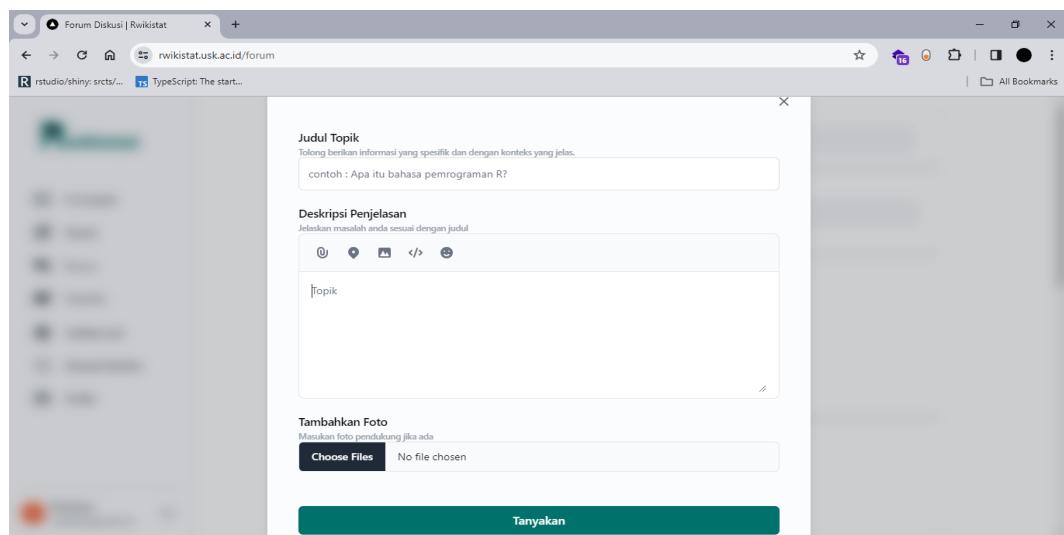
Setiap postingan bisa dilihat detailnya dengan cara meng-klik postingan yang dinginkan di halaman forum diskusi seperti yang terlihat di gambar 4.20. Jika postingan di klik, *user* akan diarahkan ke halaman detail postingan yang ditampilkan di gambar 4.21. Di halaman detail postingan, *user* bisa melihat pertanyaan yang diberikan dan gambar jika ada gambar yang dimasukkan oleh pemberi pertanyaan, komentar yang diberikan, dan *user* yang memberikan komentar. Di halaman ini *user* bisa melakukan beberapa hal, diantaranya memberikan komentar, memberikan *like*, dan memberikan *bookmark*. Untuk memberikan komentar, *user* bisa mengisi *form* yang tersedia dan menekan tombol kirim komentar. Untuk memberikan *like*, *user* bisa menekan tombol *like* yang tersedia di bagian bawah postingan. Untuk memberikan *bookmark*, *user* bisa menekan tombol *bookmark* yang tersedia di bagian bawah postingan. Untuk *user* yang sudah memberikan *like* dan *bookmark* akan berubah warna menjadi hitam. Saat tombol *like* dan *bookmark* berwarna hitam, *user* bisa menekan tombol tersebut untuk membatalkan *like* dan *bookmark* yang

telah diberikan. Terakhir ada tombol untuk meng-*copy link* dari postingan tersebut, jika tombol ini ditekan maka link dari postingan tersebut akan tersalin dan bisa di *paste* di tempat lain. *Link* yang tersalin bisa digunakan untuk melihat halaman detail dari postingan yang bersangkutan.



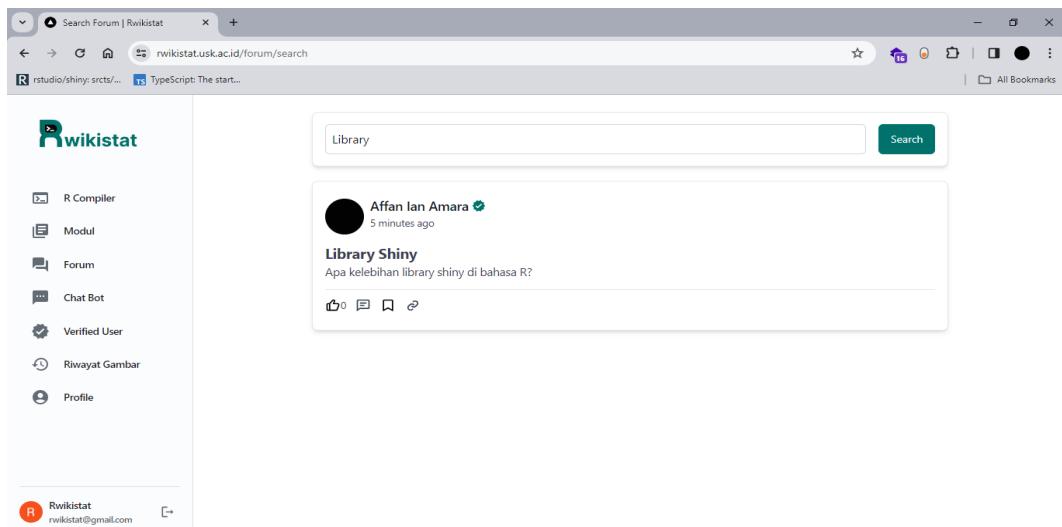
**Gambar 4.21.** Halaman detail forum diskusi

Untuk memberikan pertanyaan, *user* bisa masuk ke halaman yang ditampilkan pada gambar 4.22, di halaman ini terdapat sebuah *form* yang bisa digunakan untuk meng-*input* pertanyaan yang diinginkan. *User* bisa mengisi judul pertanyaan, topik pembahasan, dan gambar pendukung jika diperlukan. Setelah *form* di isi, *user* bisa menekan tombol kirim pertanyaan untuk mengirimkan pertanyaan yang telah diisi. Setelah pertanyaan dikirim, *user* akan diarahkan ke halaman forum diskusi dan pertanyaan yang telah dikirim akan muncul di halaman tersebut.



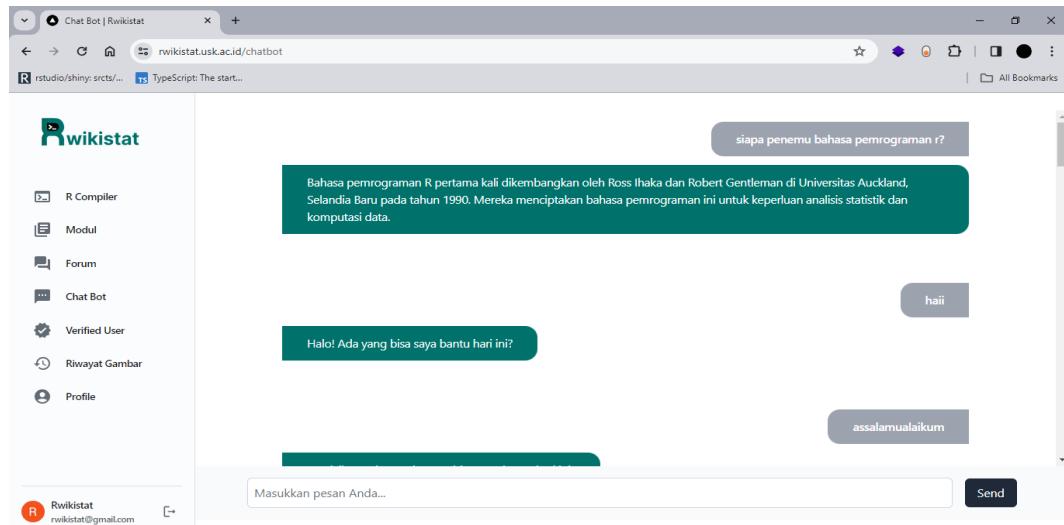
**Gambar 4.22.** Halaman tambah pertanyaan

Bagi *user* yang ingin mencari postingan tertentu, bisa dilakukan dengan masuk ke halaman search yang ditampilkan pada gambar 4.23. Di halaman ini *user* bisa mencari postingan dengan mengetikkan kata kunci yang diinginkan di *form* yang tersedia. Setelah *user* mengetikkan kata kunci, *user* bisa menekan tombol *search* untuk mencari postingan yang diinginkan. Setelah tombol *search* ditekan, postingan yang sesuai dengan kata kunci yang diinputkan akan muncul di halaman search. *User* bisa mengklik postingan yang diinginkan untuk melihat detail dari postingan tersebut.



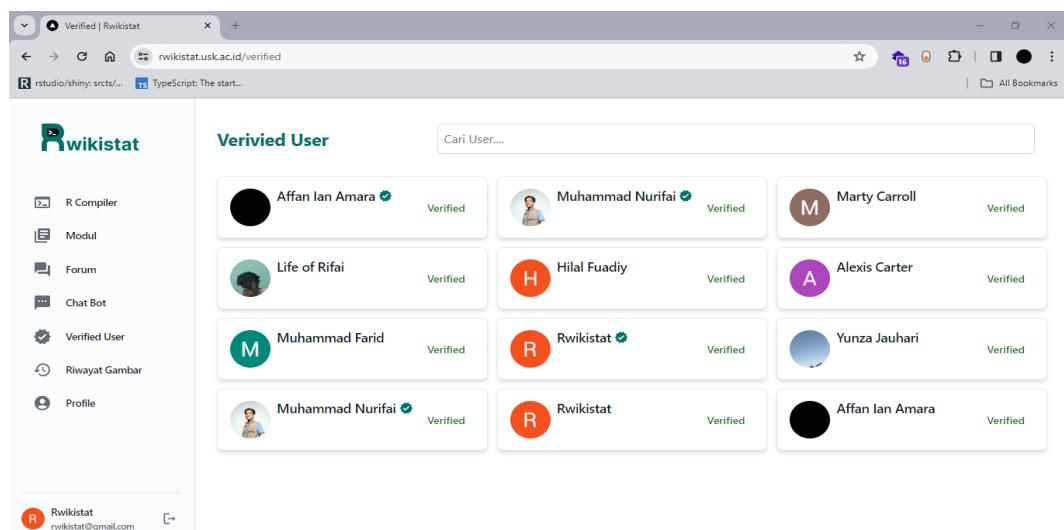
Gambar 4.23. Halaman *search*

Pada aplikasi Rwikistat juga tersedia sebuah fitur *chatbot* yang bisa digunakan *user* untuk bertanya seputar R. *User* bisa mengakses fitur ini dengan menekan tombol *chatbot* yang ada di menu *sidebar*. Setelah tombol *chatbot* ditekan, *user* akan diarahkan ke halaman *chatbot* yang ditampilkan di gambar 4.24. Di halaman ini *user* bisa mengetikkan pertanyaan yang diinginkan di *form* yang tersedia. Setelah pertanyaan dikirim, *user* akan mendapatkan jawaban dari *chatbot* yang sesuai dengan pertanyaan yang diberikan. Jawaban yang diberikan akan ditampilkan di halaman yang sama saat memberikan pertanyaan ke *chatbot*. Fitur *chatbot* ini dijalankan menggunakan pihak ketiga yaitu *OpenAi* dengan model gpt-3.5-turbo-0125.



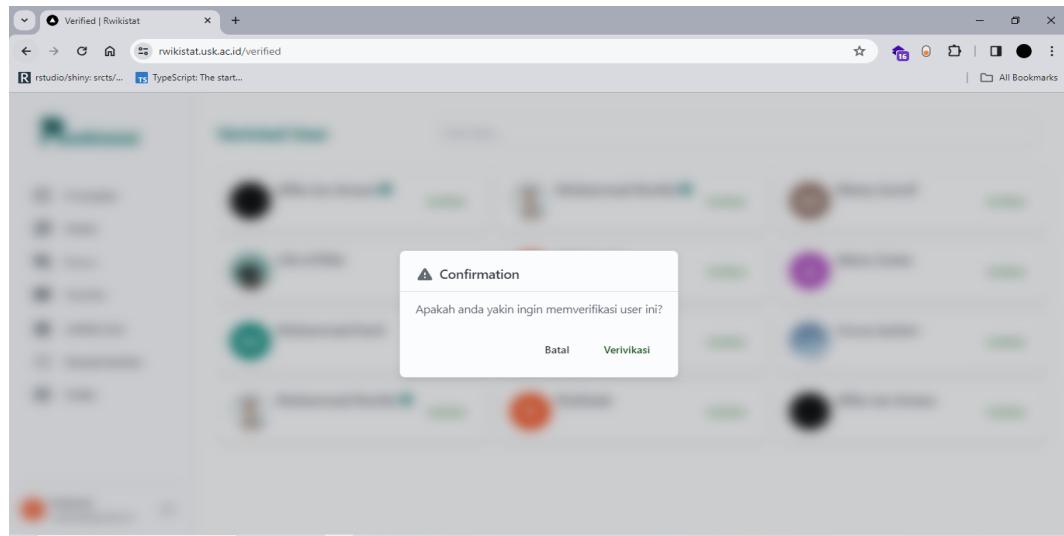
**Gambar 4.24.** Halaman *chatbot*

Selanjutnya ada halaman verifikasi *user* yang ditampilkan pada gambar 4.25, halaman ini hanya bisa diakses oleh admin. Di halaman ini terdapat daftar *user* yang terdaftar di aplikasi Rwikistat dan admin bisa memilih *user* yang akan diverifikasi terlebih dahulu, jika ada *user* yang dipilih akan muncul *popup* konfirmasi.



**Gambar 4.25.** Halaman verifikasi *user*

Pada gambar 4.26 menampilkan dialog konfirmasi apakah *user* yang dipilih benar akan diverifikasi. Jika iya admin yakin ingin memverifikasi *user* terkait, admin bisa menekan tombol verifikasi dan *user* akan diverifikasi. Jika tidak, admin bisa menekan tombol batal dan *user* tidak akan diverifikasi.



**Gambar 4.26.** Halaman konfirmasi verifikasi *user*

Pada gambar 4.27 menampilkan halaman riwayat gambar, dihalaman ini ditampilkan informasi semua gambar yang disimpan *user* sebelumnya di halaman *compiler*. Informasi yang ditampilkan berupa daftar gambar yang ditampilkan dalam bentuk nama gambar, nama gambar juga berfungsi sebagai link agar user bisa melihat gambar yang telah disimpan.

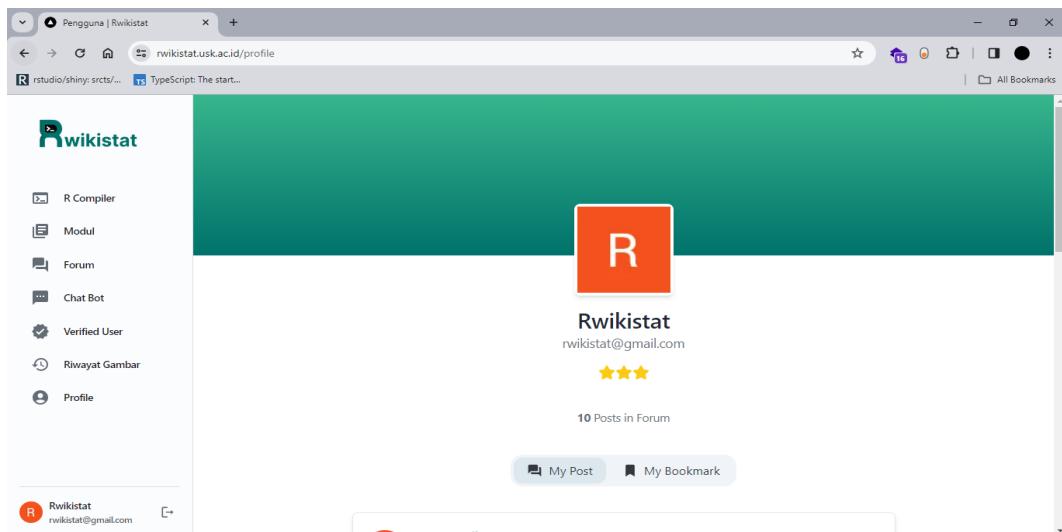
**Gambar 4.27.** Halaman riwayat gambar

Setelah *user* memilih salah satu dari daftar gambar yang ditampilkan di halaman riwayat gambar, akan muncul sebuah *popup* yang menampilkan gambar yang dipilih seperti yang ditampilkan di gambar 4.28. Pada *popup* yang ditampilkan selain menampilkan gambar, ada dua tombol yang berfungsi untuk mengunduh gambar dan hapus gambar.



**Gambar 4.28.** Halaman detail gambar

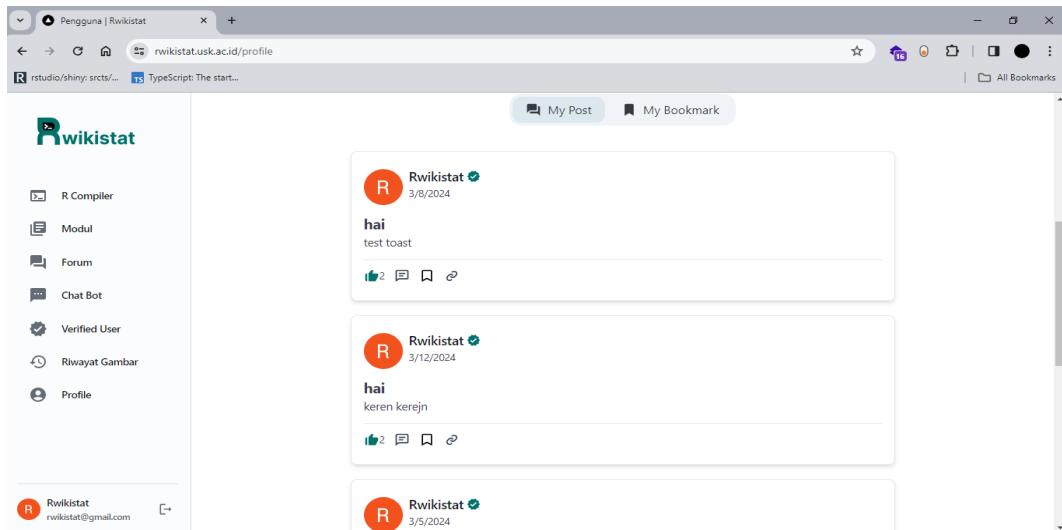
Selanjutnya ada halaman profile yang ditampilkan pada gambar 4.29, halaman ini menampilkan informasi *user* yang sedang login seperti foto profil *user*, nama, email, *score* yang dimiliki *user* dan jumlah postingan yang diberikan di forum. Di halaman ini *user* bisa melihat pertanyaan yang telah diposting dan *bookmark* yang telah diberikan. Untuk melihat pertanyaan yang telah diposting, *user* bisa menekan tombol *posted*, dan untuk melihat pertanyaan yang di *bookmark* *user* bisa menekan tombol *bookmark*. Setiap pilihan yang dipilih akan menampilkan data yang sesuai dengan yang diinginkan, baik pertanyaan yang di posting maupun pertanyaan yang di *bookmark*.



**Gambar 4.29.** Halaman *profile user*

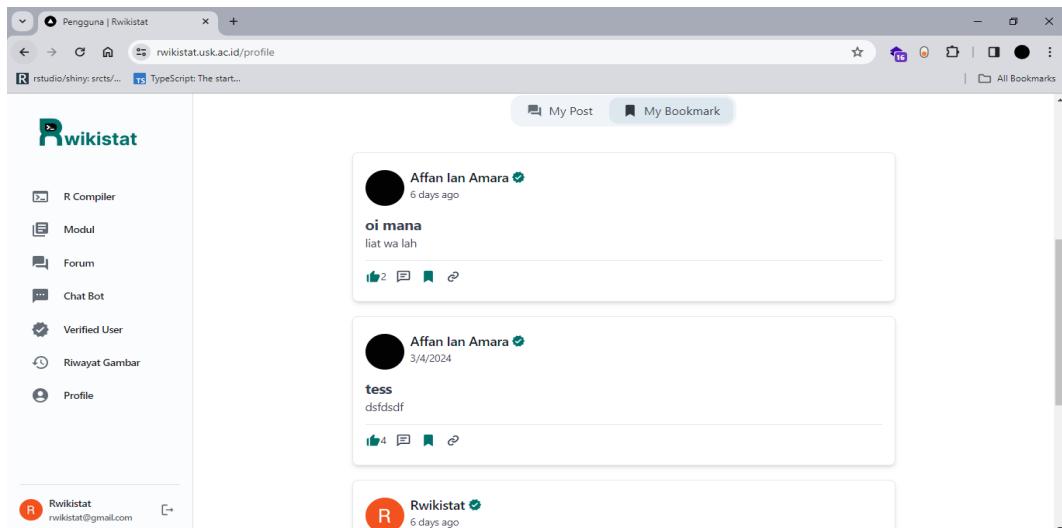
Pada gambar 4.30, ditampilkan semua postingan yang telah ditambahkan *user* di halaman forum. Data postingan *user* muncul secara *default* setiap *user* pertama

kali mengakses halaman *profile* atau bisa juga dengan menekan tombol *myPost*.



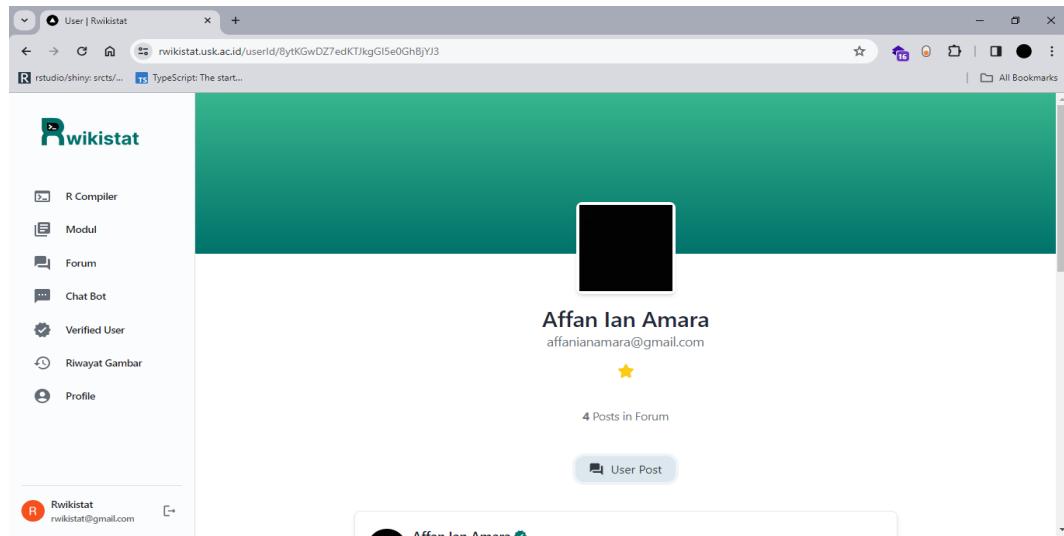
**Gambar 4.30.** Tampilan postingan *user*

Jika *user* menekan tombol *bookmark* yang ada di samping tombol *myPost*, akan ditampilkan daftar postingan forum yang di *bookmark* oleh *user* seperti yang terlihat pada gambar 4.31.



**Gambar 4.31.** Tampilan menu *bookmark user*

Selain halaman *profile* sendiri, *user* juga bisa melihat *profile* *user* lain. Untuk mengakses halaman ini bisa melalui halaman forum atau halaman *profile* sendiri yang menampilkan postingan yang di *bookmark* dengan cara meng-klik bagian nama atau foto *user* yang memposting pertanyaan. Tampilan halaman *profile* *user* lain dapat dilihat pada gambar 4.32



**Gambar 4.32.** Halaman *profile user* lain

### 4.3. PENGUJIAN APLIKASI

#### 4.3.1. Pengujian fungsionalitas menggunakan *black box testing*

Pengujian fungsionalitas menggunakan metode *black box testing* bertujuan untuk mengevaluasi kembali apakah fungsi-fungsi dalam sistem yang telah dikembangkan sesuai dengan kebutuhan dan tujuannya. Beberapa fungsi aplikasi yang telah diuji terdokumentasi dalam Tabel 4.2 dan 4.3. Dari hasil analisis Tabel 4.2 dan 4.3, dapat disimpulkan bahwa aplikasi berhasil beroperasi dengan baik, sebagaimana ditunjukkan oleh tanda "Berhasil" pada kolom hasil untuk setiap skenario.

**Tabel 4.2.** Pengujian *black box*

Modul yang di uji	Prosedur pengujian	input	output yang diharapkan	Hal yang didapat	Hasil
Halaman <i>dashboard</i>	Buka aplikasi melalui <i>browser</i>	Mengisi link website di <i>browser</i>	Menampilkan halaman <i>dashboard</i>	Menampilkan halaman <i>dashboard</i>	Berhasil
Halaman login	Masuk ke halaman login	Menekan tombol "Belajar Sekarang"	Menampilkan halaman login	Menampilkan halaman login	Berhasil

Login dengan NIM dan <i>password</i>	Masuk ke halaman <i>login</i>	Mengisi NIM dan <i>password</i>	Terbuka halaman <i>compiler</i>	Terbuka halaman <i>compiler</i>	Berhasil
Login dengan google	Masuk ke halaman <i>login</i>	Menekan tombol <i>login</i> dengan google	Terbuka halaman <i>compiler</i>	Terbuka halaman <i>compiler</i>	Berhasil
R <i>compiler</i>	Meng- <i>compiler syntax</i> R	Menekan tombol "compile" yang ada di kanan atas	Menampilkan hasil <i>syntax R</i> yang telah dikirimkan	Menampilkan hasil <i>syntax R</i> yang telah dikirimkan	Berhasil
Halaman modul	Membuka halama n modul	Menekan menu modul	Menampilkan halama n modul	Menampilkan halama n modul	Berhasil
Buka modul	Membuka modul	Menekan salah satu modul	Menampilkan halama n modul yang dipilih	Menampilkan halama n modul yang dipilih	Berhasil
Pindah modul	Mengganti modul	Membuka daftar modul dan pilih modul	Menampilkan halama n modul yang dipilih	Menampilkan halama n modul yang dipilih	Berhasil
Halaman forum diskusi	Membuka halama n forum diskusi	Menekan menu forum diskusi	Menampilkan halama n forum diskusi	Menampilkan halama n forum diskusi	Berhasil
Halaman <i>search</i>	Membuka halama n <i>search</i>	Menekan tombol <i>search</i>	Menampilkan halama n <i>search</i>	Menampilkan halama n <i>search</i>	Berhasil

<i>Search</i>	Mencari postingan	Mengisi <i>form search</i>	Menampilkan postingan yang sesuai dengan kata kunci	Menampilkan postingan yang sesuai dengan kata kunci	Berhasil
Halaman pertanyaan	Membuka halama n tambah pertanyaan	Menekan tombol tambah pertanyaan	Menampilkan halama n tambah pertanyaan	Menampilkan halama n tambah pertanyaan	Berhasil
Tambah pertanyaan	Menambahkan pertanyaan	Mengisi <i>form</i> tambah pertanyaan	Pertanyaan berhasil ditambahkan	Pertanyaan berhasil ditambahkan	Berhasil
Detail pertanyaan	Melihat detail pertanyaan	Menekan salah satu pertanyaan	Menampilkan halama n detail pertanyaan	Menampilkan halama n detail pertanyaan	Berhasil
Komentar	Memberikan komentar	Mengisi <i>form</i> komentar	Komentar berhasil ditambahkan	Komentar berhasil ditambahkan	Berhasil
<i>Like</i>	Memberikan <i>like</i>	Menekan tombol <i>like</i>	<i>Like</i> berhasil ditambahkan	<i>Like</i> berhasil ditambahkan	Berhasil
<i>Bookmark</i>	Memberikan <i>bookmark</i>	Menekan tombol <i>bookmark</i>	<i>Bookmark</i> berhasil ditambahkan	<i>Bookmark</i> berhasil ditambahkan	Berhasil
<i>unlike</i>	Membatalkan <i>like</i>	Menekan tombol <i>like</i>	<i>Like</i> berhasil dibatalkan	<i>Like</i> berhasil dibatalkan	Berhasil
<i>unbookmark</i>	Membatalkan <i>bookmark</i>	Menekan tombol <i>bookmark</i>	<i>Bookmark</i> berhasil dibatalkan	<i>Bookmark</i> berhasil dibatalkan	Berhasil
<i>copy link</i>	Meng-copy <i>link</i>	Menekan tombol <i>copy</i>	<i>link</i> tersalin	<i>link</i> tersalin	berhasil

Halaman <i>chatbot</i>	Membuka halaman <i>chatbot</i>	Menekan menu <i>chatbot</i>	Menampilkan halama <i>chatbot</i>	Menampilkan halama <i>chatbot</i>	Berhasil
<i>Chatbot</i>	Menggunakan <i>Chatbot</i>	Mengisi <i>form</i> pertanyaan	Jawaban berhasil didapatkan	Jawaban berhasil didapatkan	Berhasil
Halaman <i>profile</i>	Membuka halaman <i>profile</i>	Menekan menu <i>profile</i>	Menampilkan halama <i>n profile</i>	Menampilkan halama <i>n profile</i>	Berhasil
Melihat <i>bookmark</i>	Melihat postingan yang di <i>bookmark</i>	Menekan tombol "My Bookmark"	Menampilkan postingan yang di <i>bookmark</i>	Menampilkan postingan yang di <i>bookmark</i>	Berhasil

**Tabel 4.3.** Pengujian *black box* admin

Modul yang di uji	Prosedur pengujian	input	output yang diharapkan	Hal yang didapat	Hasil
Halaman tambah modul	Membuka halama n tambah modul	Menekan tombol tambah modul	Menampilkan halama n tambah modul	Menampilkan halama n tambah modul	Berhasil
Tambah modul	Menambahkan modul	Mengisi <i>form</i> tambah modul	Modul berhasil ditambahkan	Modul berhasil ditambahkan	Berhasil
Hapus modul	Menghapus modul	Menekan tombol hapus modul	Modul berhasil dihapus	Modul berhasil dihapus	Berhasil
Verifikasi <i>user</i>	Memverifikasi <i>user</i>	Menekan tombol verifikasi	<i>User</i> berhasil di verifikasi	<i>User</i> berhasil di verifikasi	Berhasil

#### **4.3.2. Pengujian usabilitas menggunakan metode UMUX**

Setelah menyelesaikan pengujian fungsionalitas sistem aplikasi menggunakan metode *Black-box Testing*, langkah selanjutnya adalah melakukan pengujian usabilitas. Tujuan dari pengujian usabilitas ini adalah untuk mengevaluasi pengalaman *user* saat berinteraksi dengan aplikasi. Pengujian ini bertujuan untuk mengidentifikasi potensi masalah usabilitas dalam produk aplikasi, sehingga dapat diperbaiki dan solusinya dapat ditemukan sebelum produk aplikasi diluncurkan secara publik. Proses pengujian usabilitas ini dilakukan dengan menggunakan metode *User Metrics for User Experience* (UMUX), yang dikenal sebagai metode yang singkat, sederhana, namun efektif untuk memahami pandangan umum *user* terhadap pengalaman mereka menggunakan produk aplikasi yang diuji. UMUX merupakan alternatif yang efektif dan efisien dibandingkan dengan metode *System Usability Scale* (SUS) yang memiliki 10 pertanyaan Likert. UMUX hanya terdiri dari 4 pertanyaan Likert yang melibatkan dimensi keefektifan, efisiensi, kepuasan, dan kefrustasian dalam penggunaan aplikasi. Pertanyaan-pertanyaan khusus yang diajukan mencakup 3 pertanyaan positif dan 1 pertanyaan negatif, mengenai pengalaman pengguna dalam menguji dan mengoperasikan aplikasi. *User* diminta untuk memberikan jawaban dalam skala 1 hingga 7, yang merepresentasikan dari 'sangat tidak setuju' hingga 'sangat setuju'. Berikut adalah empat pertanyaan Likert yang diajukan selama pengujian usabilitas aplikasi ini.

1. Aplikasi ini bekerja sesuai dengan kebutuhan dan ekspektasi saya.
2. Aplikasi ini susah untuk dioperasikan dan membuat saya frustasi.
3. Aplikasi ini sangat mudah dioperasikan.
4. Aplikasi ini memakan waktu yang sangat banyak ketika dioperasikan.

**Tabel 4.4.** Skenario Pengujian Usabilitas (*Usability Testing*)

<b>Test Plan Aplikasi RwikiStat</b>
<p>Metode:</p> <p>Pengujian dilakukan secara daring dan luring, dilanjutkan dengan pengisian kuesioner melalui Google Form.</p>
<p>Skenario semua <i>user</i>:</p> <ol style="list-style-type: none"><li>1. Membuka aplikasi website di browser.</li><li>2. <i>Login</i> ke aplikasi.</li><li>3. Mengisi dan menjalankan <i>syntax R</i>.</li><li>4. Menyimpan gambar.</li><li>5. Membuka modul.</li><li>6. Memilih modul yang ingin diakses.</li><li>7. Membuka forum diskusi.</li><li>8. Membuat pertanyaan.</li><li>9. Memberikan <i>like</i> atau <i>bookmark</i>.</li><li>10. Membuka halaman detail pertanyaan.</li><li>11. Memberikan komentar.</li><li>12. Membuka halaman <i>chatbot</i>.</li><li>13. Menggunakan <i>chatbot</i>.</li><li>14. Membuka halaman riwayat gambar.</li><li>15. Membuka halaman <i>profile</i>.</li><li>16. Melihat <i>bookmark</i>.</li></ol>
<p>Skenario <i>user admin</i>:</p> <ol style="list-style-type: none"><li>1. Memambahkan modul.</li><li>2. Menghapus modul.</li><li>3. Memverifikasi <i>user</i>.</li></ol>

#### 4.4. ANALISIS USABILITAS (*Usability Analysis*)

Analisis usabilitas adalah suatu proses di mana kelompok *user* atau pengujian secara sistematis menguji kualitas penggunaan suatu produk aplikasi. Proses ini bertujuan untuk mengumpulkan umpan balik dari *user* atau pengujian mengenai pengalaman mereka dalam menggunakan atau mengoperasikan produk aplikasi tersebut. Tujuannya adalah untuk mengidentifikasi potensi masalah yang mungkin timbul, mencari solusinya, dan meningkatkan pengalaman *user* pada produk aplikasi tersebut. Pada langkah ini, peneliti telah menjalankan analisis usabilitas (*Usability Analysis*) dengan melibatkan 10 *user*, yang juga dapat disebut sebagai responden. Responden tersebut terdiri dari 5 mahasiswa jurusan statistika dan 5 mahasiswa informatika. Pengujian dilaksanakan menggunakan teknik UMUX dengan skenario yang tercantum pada Tabel 4.4. Setelah itu, respon dari seluruh responden dihitung menggunakan rumus penilaian berikut:

- Setiap item ganjil akan dihitung dengan [skor pengguna- 1 (satu)] dan item

genap akan dihitung dengan [7 (tujuh)-skor pengguna].

- Setiap item skor penilaian pengguna dijumlahkan dan dibagi 24 (dua puluh empat) yang mana merupakan skor tertinggi.
- Kemudian hasil bagi tersebut dikalikan 100 (seratus).
- Kemudian dirata-ratakan hasil tersebut pada seluruh pengguna.
- Hasil dari skor UMUX kemudian dinilai baik buruknya dengan skala 0(nol )-100(seratus).
- Kemudian dengan skala tersebut tentukan sifat berdasarkan nilai sebagai berikut :
  1. Very usable (Sangat mudah digunakan): 80 (delapan puluh) sampai dengan 100(seratus).
  2. Usable (Mudah) digunakan: 60 (enam puluh) sampai dengan 79 (tujuh puluh sembilan).
  3. Borderline usable (Cukup mudah digunakan): 40 (empat puluh) sampai dengan 59 (lima puluh sembilan).
  4. Not usable (Tidak mudah digunakan): 0 (nol) sampai dengan 39 (tiga puluh sembilan).

Tabel 4.5 dibawah ini berisi interpretasi penilaian dari setiap pengguna atau responden.

**Tabel 4.5.** Hasil pengujian dan penilaian UMUX aplikasi Rwikistat

Responden	Nomor pertanyaan				Hasil Akhir
	1	2	3	4	
Mahasiswa Statistika 1	7	1	7	1	100
Mahasiswa Statistika 2	6	2	7	2	87,5
Mahasiswa Statistika 3	5	2	6	2	79,16
Mahasiswa Statistika 4	6	2	6	1	87,5
Mahasiswa Statistika 5	7	2	5	2	83,33
Mahasiswa Statistika 6	6	1	6	2	87,5
Mahasiswa Statistika 7	7	3	6	2	83,33
Mahasiswa Informatika 1	6	2	6	2	83,33
Mahasiswa Informatika 2	6	1	7	3	87,5
Mahasiswa Informatika 3	5	4	7	1	79,16
rata-rata					85,83

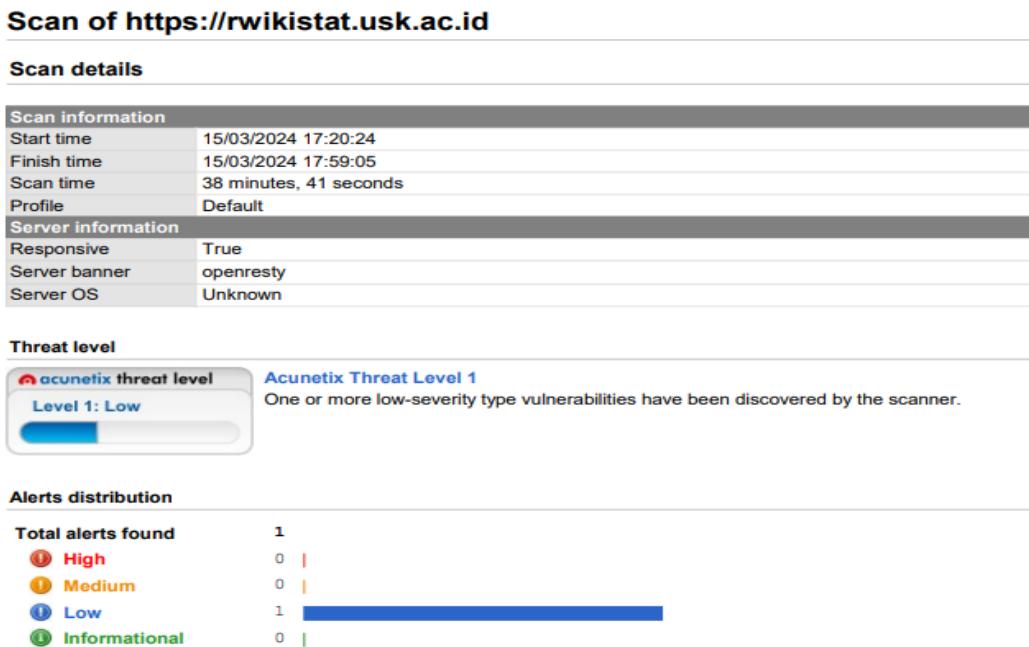
Berdasarkan informasi yang terdapat pada Tabel 4.5 di atas, dapat disimpulkan bahwa aplikasi Rwikistat memperoleh nilai rata-rata UMUX sebesar 85.83. Dengan skala penilaian dari 1 hingga 100, nilai rata-rata UMUX ini menunjukkan bahwa aplikasi Rwikistat dinilai sangat baik dan sangat layak digunakan oleh kelompok *user*. Secara khusus, nilai ini mengindikasikan bahwa aplikasi ini dapat dianggap sangat usable atau sangat mudah digunakan dan dioperasikan.

#### 4.5. PENETRATION TESTING MENGGUNAKAN APLIKASI ACUNETIX

Setelah melakukan pengujian selama sekitar 38 menit menggunakan aplikasi *Acunetix*, ditemukan bahwa *website* Rwikistat memiliki tingkat ancaman (*threat level*) 1. Pada tingkat ini, satu atau lebih kerentanan (*vulnerabilities*) dengan tingkat keparahan rendah telah terdeteksi oleh scanner. Hasil dari pengujian dapat dilihat pada gambar 4.33

Namun demikian, dari 15 URL yang diperiksa, tidak ditemukan adanya kerentanan pada URL tersebut. Hal ini menunjukkan bahwa sebagian besar bagian dari *website* tidak menunjukkan tanda-tanda kerentanan yang signifikan.

Tingkat ancaman 1 yang didapatkan menunjukkan bahwa kerentanan yang terdeteksi memiliki tingkat resiko yang rendah dan tidak menyebabkan kerusakan langsung kepada sistem. Meskipun demikian, tetap penting untuk mengawasi dan mengambil langkah-langkah pencegahan yang tepat guna mengurangi atau menghilangkan risiko terkait kerentanan tersebut.



Gambar 4.33. Hasil *penetration testing*

## **BAB V**

## **KESIMPULAN DAN SARAN**

### **5.1. KESIMPULAN**

Hasil penelitian dan pengujian terhadap aplikasi Rwikistat berbasis website yang telah dikembangkan menghasilkan beberapa kesimpulan yang dapat diuraikan sebagai berikut:

1. Forum diskusi yang dikembangkan dapat memenuhi kebutuhan pengguna pada aplikasi Rwikistat karena fitur yang dikembangkan seperti memberikan pertanyaan, komentar, *search*, *like* dan *bookmark* merupakan komponen penting untuk sebuah forum diskusi agar bisa berjalan dengan semestinya.
2. REST API telah dikembangkan untuk website dan aplikasi *mobile* Rwikistat. REST API ini dibangun menggunakan *framework* ExpressJS dan dapat digunakan untuk mengakses data dari database Rwikistat.
3. Aplikasi diuji dengan menggunakan metode *black box testing* untuk fungsionalitas dan *User Metrics for User Experience* untuk usabilitas. Hasil pengujian menunjukkan bahwa aplikasi Rwikistat yang dikembangkan sudah berjalan dengan baik dan dapat digunakan oleh pengguna.

### **5.2. SARAN**

Penelitian ini masih memiliki beberapa kelemahan yang perlu diperbaiki agar dapat mencapai tingkat yang lebih optimal. Berikut beberapa saran yang dapat diimplementasikan untuk pengembangan lebih lanjut dari penelitian ini:

1. Fitur *search* pada forum diskusi dapat dikembangkan lebih lanjut dengan filter yang lebih baik.
2. Mengembangkan R compiler yang bisa digunakan untuk menginput data dari file excel.
3. Mengintegrasikan autentikasi NIM aplikasi Rwikistat dengan SSO Universitas Syiah Kuala.
4. Meningkatkan keamanan aplikasi Rwikistat agar lebih aman dari serangan *cyber*

## DAFTAR KEPUSTAKAAN

- Ahmad, Junaid Iqbal, A. A. (2019). Model-based testing using uml activity diagrams: A systematic mapping study. *Computer Science Review*, 3(33):98–112.
- Barnum, C. M. (2020). *Usability testing essentials: ready, set... test!* Morgan Kaufmann.
- Doglio (2020). Introducing deno. pages 27–29.
- Edwin, N. M. (2018). Software frameworks, architectural and design patterns. *Journal of Software Engineering and Applications*, 2018.
- Freeman (2021). Essential typescript 4: From beginner to pro. pages 35–37.
- Galletly, R. C. (2020). Using an online discussion forum in a summative coursework assignment. *Journal of Educators Online*, 17(2).
- Hema, S. T. (2020). Scrum: An effective software development agile tool. *IOP Conf Series: Materials Science and Engineering*.
- Hizir, Rasudin, M. (2019). Analisis kepuasan pengguna aplikasi rwikistat 3.0. *Journal of Data Analysis*, 2(2):80–87.
- Hizir, M. S. (2010). Interactive statistics learning with rwikistat. *International Conference on Networking and Information Technology*, pages 557–559.
- Hizir, Edi Muttaqin, M. S. (2012). Rwikistat2.0:a web based statistical learningsystem. *International Journal of Human–Computer Interaction*, 1B:149–151.
- Jartarghar, G. R. S. (2022). React apps with server-side rendering: Next.js. *Journal of Telecommunication, Electronic and Computer Engineering*, 14(4):25–29.
- Juwanda, I. (2018). Konfigurasi rweb pada web server apache untuk aplikasi pembelajaran statistik (rwikistat) berbasis android.
- Latukolan, Achmad Arwan, M. T. A. (2019). Pengembangan sistem pemetaan otomatis entity relationship diagram ke dalam database. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 3(4):4058–4065.
- Lewis, J. R. (2018). Measuring perceived usability: The csuq, sus, and umux. *International Journal of Human–Computer Interaction*, 34(12):1148–1156.
- Loyola-Gonzalez (2019). Black-box vs. white-box: Understanding their advantages and weaknesses from a practical point of view. *Special section on deep learning: Security and forensics research advances and challenges*, 7.
- Mtshali, S. M. M. (2020). Online discussion forum: A tool to support learning in business management education. *South African Journal of Education*, 40(2).
- Ozkaya, F. E. (2020). A survey on the practical use of uml for different software architecture viewpoints. *Information and Software Technology*, 121(36).

- Prasad, S. I. (2020). Proceedings of the 2020 ACM Conference on International Computing Education Research. In *How do Graduating Students Evaluate Software Design Diagrams?*, pages 282–290.
- Raja, P. C. N. (2018). Impact of modern technology in education. *Journal of Applied and Advanced Research*, 3:33–35.
- Rajan, E. E. (2017). Web vulnerability scanners: A case study. *Cryptography and Security*.
- Rerung, Muhammad Fauzan, H. H. (2020). Website quality measurement of higher education services institution region iv using webqual 4.0 method. *International Journal of Advances in Data and Information Systems*, 1(2):89–102.
- Riihiaho, S. (2018). Usability testing. *The Wiley handbook of human computer interaction*, 1:255–275.
- Roumeliotis, N. D. T. (2023). Chatgpt and open-ai models: A preliminary review. *Future Internet 2023*, 6(15):192.
- Saks (2019). Javascript frameworks: Angular vs react vs vue. page 2.
- Sandhya, S. P. (2017). Assessment of website security by penetration testing using wireshark. *International Conference on Advanced Computing and Communication Systems (ICACCS)*.
- Santi, R. A. (2022). Black box testing with equivalence partitioning and boundary value analysis methods (study case: Academic information system of mataram university). *MIMSE 2022 (Informatics and Computer Science)*, pages 207–219.
- Sun, Elliott Zemour, A. S. (2023). Does fine-tuning gpt-3 with the openai api leak personally-identifiable information? *Arxiv Lab Machine Learning, Computation and Language*.

## **Lampiran 1**

### **Lampiran 1. Foto Pengujian Usability**



## Lampiran 2

### Lampiran 2. Data Hasil Usability Testing

Timestamp	Nama	Jurusan	Aplikasi ini bekerja sesuai	Aplikasi ini susah untuk	Aplikasi ini sangat mudah	Aplikasi ini memakan waktu
12/01/2024 17:08:51	Norman Khalis	Statistika	7	1	7	1
12/01/2024 17:09:02	Akramul Ula	Statistika	6	2	7	2
12/01/2024 17:09:49	Cut Videla Theodora	Statistika	5	2	6	2
12/01/2024 18:10:43	Naufal Tsabitul Azmi	Statistika	6	2	6	1
12/01/2024 18:20:11	Arief Rahby	Statistika	7	2	5	2
12/01/2024 18:25:55	Lutfiansyah	Statistika	6	1	6	2
12/01/2024 18:28:31	Utami	Statistika	7	3	6	2
12/01/2024 23:15:10	Reza Angga Putra	Informatika	6	1	7	3
12/01/2024 23:35:48	Daffaq Syafiq	Informatika	5	4	7	3
12/01/2024 23:53:44	Yunza Jauhari	Informatika	6	2	6	2

## **Lampiran 3**

### **Lampiran 3. Biodata Penulis**

#### **BIODATA**

1. Nama : Affan Ian Amara
2. Tempat, tanggal lahir : Asir-Asir Atas, 9 April 2001
3. Alamat : Asir - Asir, Lut Tawar, Aceh Tengah
4. Nama Ayah : Qazwin Sofyan
5. Pekerjaan Ayah : Petani
6. Nama Ibu : Aida Fitri A.
7. Pekerjaan Ibu : Pegawai Negeri Sipil
8. Alamat Orang Tua : Asir - Asir, Lut Tawar, Aceh Tengah
9. Riwayat Pendidikan :

Jenjang	Nama Sekolah	Bidang Studi	Tempat	Tahun Ijazah
SD	SD N 7 LUT TAWAR	-	ACEH TENGAH	2013
SMP	SMP N 1 TAKENGON	-	ACEH TENGAH	2016
SMA	SMA N 1 TAKENGON	IPA	ACEH TENGAH	2019

Darussalam, 2 April 2024  
Yang Menyatakan,

Affan Ian Amara  
1908107010038