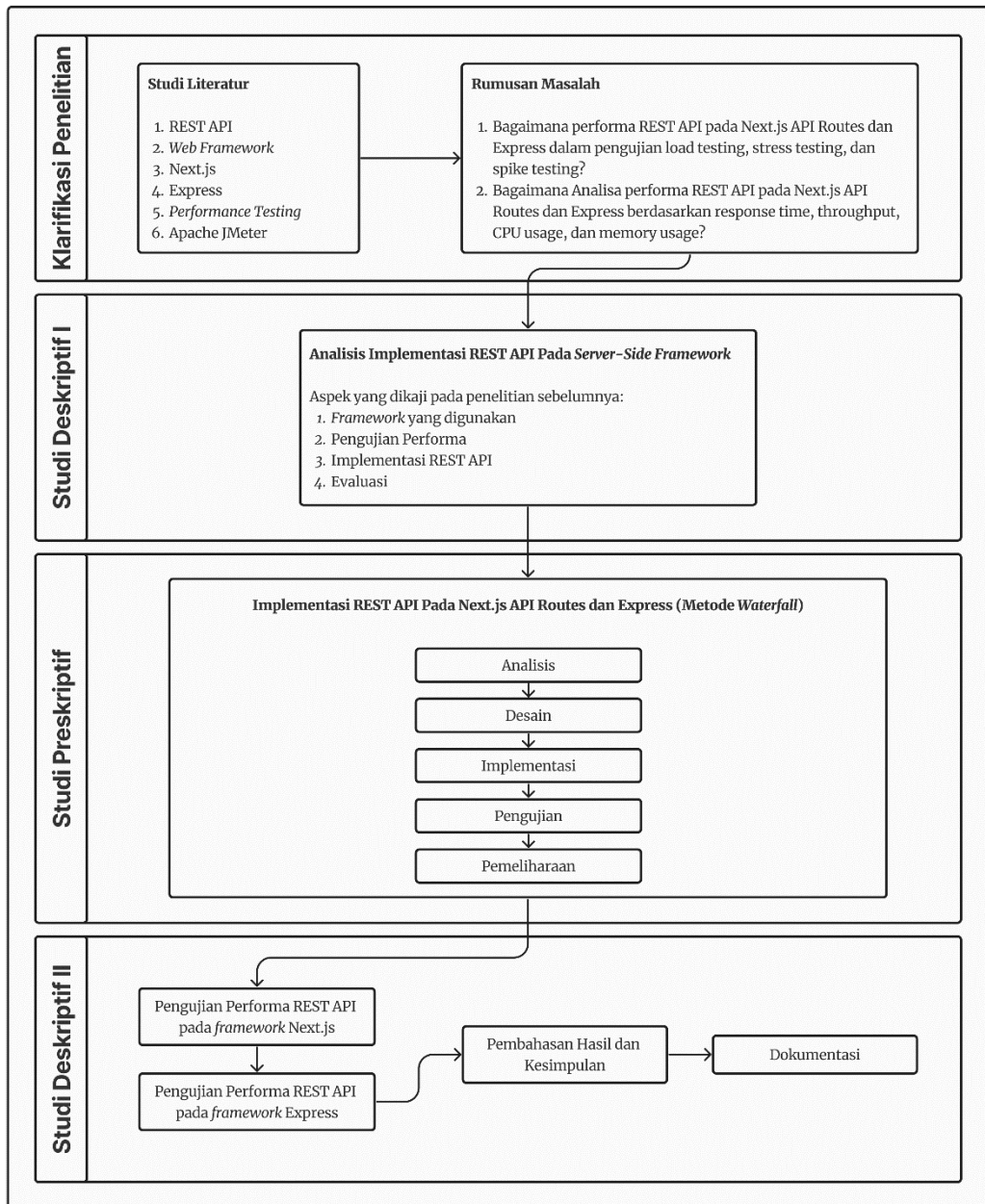


BAB III METODE PENELITIAN

3.1 Desain Penelitian

Dalam penelitian ini, diterapkan *Design Research Methodology* atau DRM sebagai metode penelitian. Adapun desain penelitian yang akan digunakan dalam penelitian ini dapat dilihat pada Gambar 3.1.



Gambar 3.1 Desain Penelitian

Jagtap dan Jagtap (2015) menjelaskan bahwa DRM terdiri dari empat tahap, yaitu (1) Klarifikasi Penelitian, (2) Studi Deskriptif I, (3) Studi Preskriptif, dan (4) Studi Deskriptif II. Metode ini dipilih oleh karena DRM menyediakan kerangka kerja untuk melakukan penelitian (Lattanzio dkk., 2019), relevan untuk digunakan dalam penelitian yang dilakukan oleh penulis.

3.1.1 Klarifikasi Penelitian

Tahap pertama yang dilakukan penulis dalam penelitian ini adalah Klarifikasi Penelitian. Pada tahap ini dilakukan studi literatur untuk mendapatkan pemahaman mengenai analisis performa yang merupakan topik penelitian dalam penelitian ini. Studi literatur yang dilakukan penulis merupakan studi literatur mengenai analisis *performance* implementasi REST API pada *framework server-side*, seperti Express, ASP.NET, Ktor, dll. Beberapa teori lainnya yang penulis dapatkan dari studi literatur yang dilakukan diantaranya adalah REST API, *Web Framework*, Next.js, Express, *Performance Testing*, *Apache JMeter*, dan lain-lain. Teori-teori yang penulis dapatkan bersumber dari jurnal, artikel ilmiah, artikel di internet, dokumentasi resmi, buku elektronik, serta lain sebagainya. Penjelasan dari teori yang didapatkan penulis dapat dilihat pada Bab II. Setelah penulis melakukan kajian teori-teori yang telah disebutkan sebelumnya, maka terbentuklah rumusan masalah berdasarkan studi literatur yang telah dilakukan.

3.1.2 Studi Deskriptif I

Tahapan selanjutnya yang dilakukan penulis adalah Studi Deskriptif I. Pada tahap ini, dilakukan analisis dan kajian penelitian-penelitian mengenai analisis *performance* yang telah dilakukan sebelumnya. Tahapan ini bertujuan untuk memperdalam serta memperkuat pemahaman penulis terkait topik yang diteliti oleh penulis. Analisis dan kajian dilakukan dari beberapa aspek, seperti *framework* yang digunakan, pengujian performa yang dilakukan, REST API yang diimplementasikan, dan evaluasi yang dilakukan untuk menganalisis performa dari *framework* yang diuji. Analisis dan kajian penelitian terdahulu dapat memberikan pemahaman yang kuat kepada penulis mengenai analisis *performance* implementasi REST API pada *framework server-side*.

3.1.3 Studi Preskriptif

Setelah melakukan Studi Deskriptif I, tahapan berikutnya yang dilakukan oleh penulis adalah Studi Preskriptif. Pada tahap ini, diimplementasikan REST API pada *framework server-side* yang akan diuji, yaitu Next.js dan Express. Metode pengembangan perangkat lunak yang digunakan oleh penulis adalah metode *waterfall*. Christanto dan Singgalen (2023) menjelaskan bahwa metode *waterfall* memiliki lima tahapan, yaitu (1) Analisis, (2) Desain, (3) Implementasi, (4) Pengujian, dan (5) Pemeliharaan. Berdasarkan tahapan-tahapan metode *waterfall* yang telah disebutkan sebelumnya, langkah-langkah tersebut akan diikuti oleh penulis. Berikut adalah penjelasan dari tahapan metode *waterfall* yang akan dilakukan oleh penulis dalam penelitian ini:

1. Analisis

Pada tahap pertama ini dilakukan analisis berdasarkan penelitian sebelumnya untuk merancang alur dari REST API yang akan diimplementasikan nanti. Analisis kebutuhan fungsional maupun non-fungsional akan dilakukan penulis dalam pengembangan perangkat lunak yang akan dilakukan, yang dalam hal ini adalah implementasi REST API pada *framework* Next.js dan Express. Selain itu, pada tahap ini juga ditentukan batasan dari REST API yang diimplementasikan.

2. Desain

Tahap selanjutnya setelah melakukan analisis adalah tahapan desain. Pada tahap ini, dilakukan perancangan REST API yang nantinya akan diimplementasikan pada *framework* Next.js dan Express. Perancangan ini didasarkan pada analisis yang telah dilakukan pada tahapan sebelumnya.

3. Implementasi

Setelah melakukan perancangan, tahap selanjutnya adalah implementasi. Pada tahap implementasi, rancangan yang telah dibuat pada tahap sebelumnya akan diimplementasikan. Implementasi REST API pada *framework* Next.js dan Express akan dilaksanakan berdasarkan rancangan yang telah dibuat.

4. Pengujian

Tahap berikutnya setelah REST API diimplementasikan pada *framework* Next.js dan Express adalah pengujian. *Endpoint-endpoint* yang telah

diimplementasikan pada Next.js dan Express akan diuji oleh penulis. Pengujian ini dilakukan dengan tujuan untuk memastikan bahwa REST API yang diimplementasikan berjalan sesuai dengan rancangan awal. Selain itu, pengujian juga dilakukan untuk memastikan kebutuhan perangkat lunak yang telah dibuat terpenuhi.

5. Pemeliharaan

Pada tahap pemeliharaan, dilakukan pemeliharaan terhadap perangkat lunak yang telah dikembangkan. Pemeliharaan yang dilakukan diantaranya adalah melakukan perbaikan pada perangkat lunak jika terdapat kesalahan atau *bug* yang ditemukan. Pada tahap ini juga dilakukan peningkatan perangkat lunak jika diperlukan.

3.1.4 Studi Deskriptif II

Setelah tahap Studi Preskriptif dilakukan, maka tahap selanjutnya yang dilakukan adalah Studi Deskriptif II. Pada tahap ini, pengujian performa pada REST API yang telah diimplementasikan pada Next.js dan Express akan dilakukan. Pengujian performa ini dilakukan untuk mengetahui performa implementasi REST API pada *framework* Next.js dan Express, yang merupakan tujuan dari penelitian ini. Pengujian performa akan dilakukan menggunakan aplikasi JMeter. Pengujian performa yang akan diujikan pada *framework* Next.js dan Express adalah *load*, *stress*, dan *spike testing*.

Selanjutnya, hasil dari pengujian performa yang dilakukan akan dianalisis dan dievaluasi oleh penulis. Hasil pengujian performa kemudian akan dianalisis berdasarkan metrik yang telah ditentukan pada rumusan masalah. Metrik yang akan digunakan oleh penulis adalah *response time*, *throughput*, *CPU usage*, dan *memory usage*. Hasil dari analisis yang dilakukan selanjutnya digunakan untuk membuat kesimpulan yang menjawab rumusan masalah yang tercantum pada Bab I. Selain itu, saran akan diberikan untuk penelitian selanjutnya agar penelitian ini dapat dilanjutkan dan ditingkatkan menjadi lebih baik.

3.2 Alat dan Bahan Penelitian

3.2.1 Alat Penelitian

Alat yang digunakan oleh penulis dalam penelitian ini terdiri dari perangkat keras (*hardware*) dan perangkat lunak (*software*). Adapun perangkat keras dan perangkat lunak yang digunakan oleh penulis adalah sebagai berikut:

- a. Perangkat keras:
 1. Prosesor: AMD Ryzen 7 5800H @ 3.20 GHz
 2. RAM: 16 GB DDR4
 3. Penyimpanan: SSD 1,5 TB
- b. Perangkat lunak:
 1. Windows 11 Home Single Language
 2. Visual Studio Code
 3. Google Chrome
 4. Node.js
 5. Node Package Manager (NPM)
 6. Next.js
 7. Express
 8. MongoDB (*database*)
 9. Apache JMeter
 10. Postman

MongoDB digunakan sebagai *database* pada penelitian ini untuk mensimulasikan REST API yang berinteraksi dengan *database*. Alasan mengapa MongoDB digunakan pada penelitian ini oleh karena MongoDB merupakan *database* tipe NoSQL (Erlandsson dan Remes, 2020), sehingga menjadi pembeda dengan penelitian-penelitian sebelumnya yang menggunakan *database* tipe SQL (Karlsson, 2021; Greiff dan Johansson, 2019). Penelitian-penelitian sebelumnya memberikan saran untuk penelitian selanjutnya agar menggunakan *database* yang berbeda (Greiff dan Johansson, 2019), seperti *database* NoSQL atau *non-relational* (Karlsson, 2021). Oleh karena itulah MongoDB digunakan sebagai *database* pada penelitian ini.

3.2.2 Bahan Penelitian

Bahan penelitian yang digunakan oleh penulis dalam penelitian ini berupa jurnal, artikel di internet, dokumentasi resmi dari teknologi yang digunakan, buku elektronik, serta lain sebagainya yang menunjang penulis dalam melakukan penelitian dan mengimplementasikan REST API pada Next.js API Routes dan Express.

3.3 Instrumen Penelitian

Instrumen penelitian yang digunakan oleh penulis dalam penelitian ini adalah alat *performance testing* yang bernama JMeter. JMeter digunakan untuk melakukan pengujian performa pada REST API yang diimplementasikan pada Next.js API Routes dan Express. Parameter *performance testing* yang digunakan penulis dalam penelitian ini adalah *response time*, *throughput*, *CPU usage*, dan *memory usage*.

3.4 Analisis Data

Parameter yang digunakan pada penelitian ini adalah *response time*, *throughput*, *CPU usage*, dan *memory usage*. Parameter tersebut akan digunakan untuk menilai performa dari *framework* yang diuji. Berikut adalah kriteria pengujian yang digunakan penelitian ini:

Tabel 3.1

Kriteria Pengujian. Sumber: Google (2021), Hadinata dan Stianingsih (2024), dan Siahaan dan Wijaya (2024)

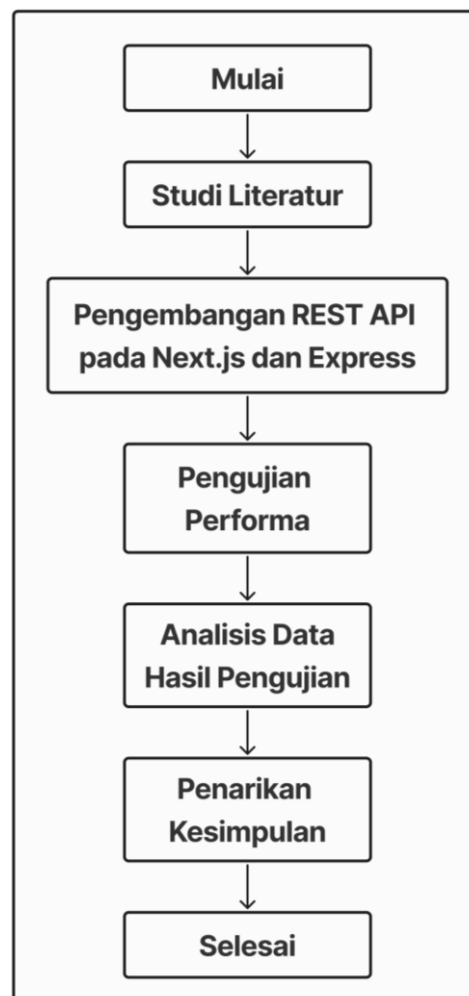
Metrik	Kriteria
<i>Response Time</i>	< 200ms
<i>CPU Usage</i>	< 75%
<i>Memory Usage</i>	< 75%

Kriteria diatas ditetapkan berdasarkan studi literatur yang telah dilakukan. Kriteria untuk *response time* didapatkan dari rekomendasi yang dikemukakan oleh Google (2021). Kriteria untuk *CPU usage* dan *memory usage* ditetapkan demikian oleh karena kriteria tersebut juga telah digunakan oleh penelitian terdahulu, yakni penelitian yang dilakukan oleh Hadinata dan Stianingsih (2024) dan Siahaan dan

Wijaya (2024). Sedangkan untuk *throughput*, tidak ditetapkan kriteria untuk nilai *request per second* oleh karena penelitian terdahulu tidak menetapkan kriteria untuk *throughput*, namun hanya berfokus pada melakukan perbandingan nilai *throughput* di antara *framework* yang diuji. Penelitian menetapkan bahwa *framework* dengan *throughput* yang lebih besar merupakan *framework* yang baik dalam menangani permintaan dalam satu waktu.

3.5 Prosedur Penelitian

Prosedur penelitian yang akan dilakukan oleh penulis didasarkan pada desain penelitian yang telah ditetapkan sebelumnya. Prosedur penelitian yang akan digunakan oleh penulis dalam penelitian ini dapat dilihat pada Gambar 3.2 berikut:



Gambar 3.2 Prosedur Penelitian