

**ANALISIS PERBANDINGAN PERFORMA API *WEB SERVICE*  
MENGUNAKAN *FRAMEWORK* SPRINGBOOT DAN LARAVEL**

**(Skripsi)**

**Oleh:**

**Rizky Yulianto**

**NPM 1815061028**



**FAKULTAS TEKNIK  
UNIVERSITAS LAMPUNG**

**BANDAR LAMPUNG**

**2024**

## ABSTRAK

### ANALISIS PERBANDINGAN PERFORMA API *WEB SERVICE* MENGUNAKAN *FRAMEWORK* SPRINGBOOT DAN LARAVEL

Oleh  
Rizky Yulianto

Dengan berkembangnya teknologi informasi dan komunikasi yang semakin meningkat dibutuhkan cara agar dapat mengakses informasi yang diinginkan dengan cepat, oleh karena itu dibuat aplikasi *web service*. *Web service* adalah sebuah aplikasi perangkat lunak yang dapat memungkinkan untuk memfasilitasi berbagai kepentingan pengguna. *Web service* menggunakan *Representational State Transfer Application Interface* atau REST API sebagai sarana komunikasi yang umum digunakan untuk mengirimkan data dengan metode *Create, Read, Update* dan *Delete* (CRUD). Untuk memudahkan pengembangan *web service* terdapat berbagai macam *framework* contohnya adalah Spring Boot untuk bahasa Java dan Laravel untuk bahasa PHP. Untuk memilih *framework* yang tepat dalam pengembangan *web service* performa waktu respon dari kedua *framework* ini harus diuji dengan cara mensimulasikan beban penggunaan berupa 10000 *request* untuk tiap metode *Post, Get, Update, dan Delete*. Dari pengujian yang dilakukan menggunakan Apache Jmeter diambil jumlah respon dan waktu respon dari kedua *web service* yang dibangun menggunakan Spring Boot dan Laravel. Didapatkan bahwa pada 10000 *request* dengan metode *Post, Put, Get, dan Delete* berturut-turut Spring Boot memiliki keunggulan rata rata waktu respon yaitu 44,8ms, 45,37ms, 109,5ms, 42,ms dibandingkan Laravel yaitu 291,98ms, 301,2ms, 426,54ms, 273,06ms.

Kata kunci: Web Service, REST API, Spring Boot, Laravel, Apache Jmeter, Waktu Respon, CRUD , Load Testing.

## **ABSTRACT**

### ***Performance difference analysis of web service api using Spring Boot and Laravel Framework***

**By  
Rizky Yulianto**

*With the increasing development of information and communication technology, there is a need for ways to quickly access the desired information. Therefore, a web service application was created. A web service is a software application that can facilitate various user interests. Web services use the Representational State Transfer Application Interface or REST API as a commonly used communication tool to send data using the Create, Read, Update, and Delete (CRUD) method. To make it easier to develop web services, there are various frameworks. For example, Spring Boot is used for the Java language, and Laravel is used for the PHP language. To choose the right framework for web service development, the response time performance of these two frameworks must be tested by simulating the usage load in the form of 10,000 requests for each Post, Get, Update, and Delete method. From tests carried out using Apache JMeter, the number of responses and response times were taken from the two web services built using Spring Boot and Laravel. It was found that for 10,000 requests using the Post, Put, Get, and Delete methods respectively, Spring Boot had an average response time advantage of 44.8ms, 45.37ms, 109.5ms, and 42ms compared to Laravel, namely 291.98ms, 301.2ms, 426.54ms, and 273.06ms.*

*Keywords : Web Service, REST API, Spring Boot, Laravel, Apache Jmeter, Response Time, CRUD.*

**ANALISIS PERBANDINGAN PERFORMA API *WEB SERVICE*  
MENGUNAKAN *FRAMEWORK* SPRINGBOOT DAN LARAVEL**

**Oleh**

**Rizky Yulianto**

**Skripsi**

Sebagai Salah Satu Syarat untuk Mencapai Gelar

**SARJANA TEKNIK**

**Pada**

**Program Studi Teknik Informatika**

**Jurusan Teknik Elektro**

**Fakultas Teknik Universitas Lampung**



**FAKULTAS TEKNIK  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG**

**2024**

Judul : **ANALISIS PERBANDINGAN PERFORMA API WEB SERVICE MENGGUNAKAN FRAMEWORK SPRINGBOOT DAN LARAVEL**

Nama Mahasiswa : Rizky Yulianto

Nomor Pokok Mahasiswa : 1815061028

Program Studi : Teknik Informatika

Fakultas : Teknik

**MENYETUJUI**

1. Komisi Pembimbing



**Dr. Eng. Ir. Mardiana, S.T., M.T., I.P.M.**

NIP. 197203161999032002



**Rio Arlestia P, S.Kom., M.T.I.**

NIP. 198603232019031013



2. Mengetahui

Ketua Jurusan  
Teknik Elektro



**Herlinawati, S.T., M.T.**

NIP. 197103141999032001

Ketua Program Studi  
Teknik Informatika



**Yessi Mulyani, S.T., M.T.**

NIP. 197312262000122001

MENGESAHKAN

1. Tim Penguji

Ketua : Dr. Eng. Ir. Mardiana, S.T., M.T., I.P.M.



Sekretaris : Rio Ariestia Pradipta, S.Kom., M.T.I



Penguji : Ir. Gigih Forda Nama, S.T., M.T.I., I.P.M.



2. Dekan Fakultas Teknik



Dr. Eng. Ir. Helmi Fitriawan, S.T., M.Sc.

NIP. 197509282001121002



Tanggal Lulus Ujian Skripsi: 22 Desember 2023

## SURAT PERNYATAAN

Saya yang bertandatangan dibawah ini, menyatakan bahwa skripsi saya yang berjudul "ANALISIS PERBANDINGAN PERFORMA API *WEB SERVICE* MENGGUNAKAN *FRAMEWORK* SPRINGBOOT DAN LARAVEL" dengan ini menyatakan bahwa skripsi saya dibuat oleh saya sendiri. Semua hasil yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti bahwa skripsi ini merupakan salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan hukum atau akademik yang berlaku.

Bandar Lampung, 17 Februari  
2024

Pembuat Pernyataan,



Rizky Yulianto

NPM 1815061028

## RIWAYAT HIDUP



Penulis bernama Rizky Yulianto yang merupakan anak ketiga dari pasangan Yaumuddin dan Febriyani. Penulis lahir di Kota Bandar Lampung pada tanggal 13 Juli 2000. Penulis telah menyelesaikan sekolah dasar di SD Negeri 1 Bandar Lampung pada tahun 2012. SMP Negeri 14 Bandar Lampung pada tahun 2015 dan SMA Negeri 3 Bandar Lampung pada tahun 2018. Penulis melanjutkan pendidikan di Fakultas Teknik Jurusan Teknik Elektro Program Studi Teknik Informatika Universitas Lampung pada tahun 2018. Selama menempuh pendidikan di Jurusan Teknik Elektro Program Studi Teknik Informatika Universitas Lampung, penulis juga aktif mengikuti beberapa kegiatan sebagai berikut

1. Pengurus Unit Kegiatan Mahasiswa English Society Universitas Lampung pada tahun 2019 -2021.
2. Pada bulan Agustus-September 2021, penulis mengikuti Kuliah Kerja Nyata selama 40 hari di Kelurahan Way Tataan, Kecamatan Teluk Betung Timur, Kota Bandar Lampung.
3. Menjadi peserta program belajar kampus merdeka di Binar Academy dengan learning path Java Backend



## **PERSEMBAHAN**

*Bismillahirrahmanirrahim....*

Segala Puji syukur kepada Allah SWT berkat karunia, kesehatan, rahmat serta hidayah-Nya yang telah diberikan, shalawat teriring salam kepada Nabi Muhammad SAW, suri tauladan Akhlaqul Kharimah yang kita nantikan syafa'atnya di hari akhir kelak. Dengan segala kerendahan hati, saya persembahkan skripsi ini kepada:

Ibuku, yang telah melahirkanku, merawatku, membesarkanku, dan yang telah sepenuh hati mendidikku.

Ayahku tercinta, yang telah membesarkanku dengan seluruh kasih dan sayangnya, memberikan pengetahuannya, dan selalu mendukung serta mendoakan untuk keberhasilanku.

Serta Adikku tersayang

Serta, almamater yang saya sangat banggakan

**UNIVERSITAS LAMPUNG**

## SANWACANA

Segala puji hanya bagi Allah SWT, Tuhan semesta alam yang Maha Pengasih lagi Maha Penyayang, atas limpahan rahmat, taufik serta hidayah-Nya sehingga penulis dapat menyelesaikan penulisan skripsi yang berjudul: *ANALISIS PERBANDINGAN PERFORMA API WEB SERVICE MENGGUNAKAN FRAMEWORK SPRINGBOOT DAN LARAVEL* Sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik pada Fakultas Teknik Universitas Lampung. Dalam penyelesaian skripsi ini, penulis mendapatkan banyak bantuan, bimbingan, saran dan dukungan dari segenap pihak, baik secara langsung maupun tidak langsung sehingga penyusunan skripsi ini berjalan dengan baik. Maka pada kesempatan ini penulis mengucapkan terima kasih kepada:

1. Bapak Dr. Eng. Helmy Fitriawan, S.T., M.Sc selaku Dekan Fakultas Teknik Universitas Lampung.
2. Ibu Herlinawati, S.T., M.T. selaku Ketua Jurusan Teknik Elektro Universitas Lampung.
3. Ibu Dr. Eng. Ir. Mardiana, S.T., M.T., IPM selaku Pembimbing Utama, yang telah bersedia meluangkan waktu untuk memberikan pengarahan dan bimbingan dalam pembuatan skripsi ini hingga selesai.
4. Ibu Yessi Mulyani, S.T., M.T. selaku Ketua Program Studi Teknik Informatika Jurusan Teknik Elektro Universitas Lampung.
5. Bapak Rio Ariestia Pradipta, S.Kom., M.TI. selaku Pembimbing Kedua, yang telah bersedia meluangkan waktu untuk memberikan pengarahan dan bimbingan dalam pembuatan skripsi ini hingga selesai.
6. Pak Wahyu Eko Sulistiono, S.T., M.Sc, selaku Pembimbing Akademik yang telah bersedia meluangkan waktu untuk memberikan pengarahan dan bimbingan selama menempuh pendidikan Program Studi Teknik Informatika Universitas Lampung.

7. Bapak Ir. Gigih Forda Nama, S.T., M.T.I., I.P.M selaku Penguji yang memberikan banyak arahan kepada penulis.
8. Seluruh Dosen Program Studi Teknik Informatika yang telah membagikan ilmunya kepada penulis.
9. Teman-teman The Real Wibu dan Last Brain Cells yang selalu membantu dalam menyusun skripsi.
10. Seluruh teman-teman Teknik Informatika Angkatan 2018 selaku teman kelas.

Semoga Allah SWT membalas segala bentuk kebaikan hati dan jasa yang telah kalian berikan kepada saya. Saya menyadari meskipun skripsi ini sudah disusun dengan sebaik mungkin, skripsi ini masih terdapat kekurangan dan masih jauh dari kata sempurna, namun saya sangat berharap melalui skripsi ini akan memberikan manfaat bagi siapapun yang membacanya dan bagi penulis dalam mengembangkan dan mengamalkan ilmu pengetahuan yang telah ditempuh selama ini.

Bandar Lampung, 17 Februari 2024

Penulis,

A handwritten signature in black ink, appearing to be 'Rizky Yulianto', written over a light blue horizontal line. The signature is stylized and cursive.

**Rizky Yulianto**

## DAFTAR ISI

DAFTAR ISI .....	i
Daftar Tabel.....	iii
Daftar Gambar.....	iv
I PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	4
1.3 Tujuan Penelitian.....	4
1.4 Manfaat Penelitian.....	4
1.5 Batasan Masalah.....	5
1.6 Sistematika Penulisan.....	5
II. TINJAUAN PUSTAKA .....	7
2.1 <i>Web Service</i> .....	7
2.2 Hyper Text Transfer Protocol (HTTP) .....	7
2.3 Rest API .....	7
2.4 <i>Framework</i> .....	8
2.5 Spring Boot .....	9
2.6 Laravel.....	9
2.7 Pengujian performa .....	10
2.8 <i>Aggregate Report jmeter</i> .....	11
2.9 XAMPP .....	11
2.10 Penelitian terkait.....	12
2.10.1 Analisis Perbandingan Performa Web Service Rest Menggunakan Framework Laravel, Django, Dan Node Js Pada Aplikasi Berbasis Website	12
2.10.2 Analisis Komparasi Performa Web Application: Studi Kasus Asp.Net Mvc Dan Asp.Net Core .....	12
2.10.3 A Performance Comparison of RESTful Applications Implemented in Spring Boot Java and MS.NET Core .....	13
2.10.4 A Comparative Investigation on Implementation of RESTful versus SOAP based Web Services .....	13
2.10.5 Comparative study on Python web frameworks: Flask and Django ..	13

2.10.6 Comparative Study of Load Testing Tools: Apache JMeter, HP LoadRunner, Microsoft Visual Studio (TFS), Siege .....	14
III. METODOLOGI PENELITIAN.....	17
3.1 Waktu dan Tempat.....	17
3.2 Alat Dan Bahan .....	18
3.2.1 Alat dan bahan penelitian .....	18
3.3 Tahapan Penelitian .....	19
1. Perancangan <i>Database</i> .....	21
2. Perancangan <i>Web Service</i> .....	22
3. Pengujian dengan Jmeter.....	26
4. Penarikan Kesimpulan dan Penulisan Laporan.....	31
IV. PEMBAHASAN .....	32
4.1 Perancangan <i>Database</i> .....	32
4.1.1 Rancangan <i>Database</i> Spring Boot .....	32
4.1.2 Rancangan Database Laravel .....	33
4.2 Perancangan Web Sistem .....	34
4.2.1 Perancangan Web Sistem dengan Spring Boot .....	34
4.2.2 Perancangan Web Sistem dengan Laravel .....	36
4.3 Pengujian dengan Jmeter.....	39
4.3.1 Pengujian Web Service framework Spring Boot.....	39
4.3.2 Pengujian Web Service framework Laravel.....	48
4.4 Penarikan Kesimpulan Dari Hasil Penelitian.....	56
V. KESIMPULAN.....	60
5.1 Kesimpulan .....	60
5.2 Saran.....	61
Daftar Pustaka .....	62

## Daftar Tabel

Tabel 2.1 Tabel ringkasan penelitian.....	14
Tabel 3.1 Jadwal Penelitian.....	17
Tabel 3.2 Alat dan bahan penelitian .....	18
Tabel 4.1 data hasil pengujian metode POST web service Spring Boot.....	42
Tabel 4.2 data hasil pengujian metode PUT <i>web service</i> Spring Boot .....	44
Tabel 4.3 data hasil pengujian metode GET <i>web service</i> Spring Boot .....	45
Tabel 4.4 data hasil pengujian metode DELETE <i>web service</i> Spring Boot.....	47
Tabel 4.5 data hasil pengujian metode POST <i>web service</i> Laravel .....	50
Tabel 4.6 data hasil pengujian metode PUT <i>web service</i> Laravel.....	52
Tabel 4.7 data hasil pengujian metode GET pada web service Laravel.....	53
Tabel 4.8 data hasil pengujian metode DELETE pada web service Laravel .....	55

## Daftar Gambar

Gambar 1.1 Diagram batang 16 <i>framework</i> populer tahun 2023.....	3
Gambar 2.1 Diagram Rest API.....	8
Gambar 2.2 Diagram <i>framework</i> Spring Boot[13] .....	9
Gambar 2.3 Diagram <i>Framework</i> Laravel .....	10
Gambar 2.4 Logo Apache Jmeter.....	11
Gambar 2.5 Logo XAMPP.....	11
Gambar 3.1 Alur Penelitian.....	20
Gambar 3.2 Tabel Database Pengujian .....	21
Gambar 3.3 class diagram web service .....	22
Gambar 3.4 Halaman Muka Spring Initializr.....	23
Gambar 3.5 Proyek yang sudah di ekstrak.....	24
Gambar 3.6 Composer berhasil di instalasi.....	25
Gambar 3.7 perintah untuk membuat proyek framework laravel dengan nama LaravelBenchmark .....	25
Gambar 3.8 <i>output</i> saat instalasi berhasil.....	26
Gambar 3.9 Rancangan Pengujian .....	27
Gambar 3.10 Tampilan antarmuka aplikasi Jmeter .....	28
Gambar 3. 11 Penambahan <i>Thread Group</i> .....	28
Gambar 3.12 <i>Virtual User</i> .....	29
Gambar 3.13 Menambahkan <i>http request</i> .....	30
Gambar 3.14 <i>Http Request</i> .....	30
Gambar 4.1 <i>Class Entity</i> Pengguna Spring Boot .....	32
Gambar 4.2 Class Pengguna dalam Proyek Laravel .....	33
Gambar 4.3 <i>Interface Repository</i> Pengguna .....	34
Gambar 4.4 <i>Class Controller</i> Pengguna .....	35
Gambar 4.5 <i>Method</i> registerPengguna .....	35
Gambar 4.6 <i>Method</i> deletePengguna .....	36
Gambar 4.7 <i>method</i> updatePengguna .....	36
Gambar 4.8 file routes untuk menyimpan <i>endpoint</i> .....	37
Gambar 4.9 <i>method</i> index .....	37
Gambar 4.10 <i>method</i> registerPengguna .....	38
Gambar 4.11 <i>method</i> updatePengguna.....	38
Gambar 4.12 <i>method</i> hapusPengguna.....	39
Gambar 4.13 <i>Web service</i> Spring Boot berhasil dijalankan .....	40
Gambar 4.14 Request POST .....	41
Gambar 4.15 HTTP Header Manager .....	42
Gambar 4.16 Request <i>PUT</i> .....	43
Gambar 4.17 <i>Counter</i> pada request <i>PUT</i> Spring Boot.....	44
Gambar 4.18 <i>Request GET</i> Spring Boot .....	45
Gambar 4.19 Request DELETE.....	46
Gambar 4.20 Counter pada Method Delete.....	47

Gambar 4.21 <i>Web service</i> Laravel berhasil dijalankan .....	48
Gambar 4. 22 <i>Request POST</i> Laravel .....	49
Gambar 4.23 <i>HTTP Header Manager</i> .....	50
Gambar 4.24 <i>Request PUT</i> Laravel .....	51
Gambar 4.25 <i>Counter</i> pada <i>request PUT</i> Laravel.....	52
Gambar 4.26 <i>Request GET</i> Laravel .....	53
Gambar 4.27 <i>Request DELETE</i> .....	54
Gambar 4.28 <i>Counter</i> pada <i>Method Delete</i> .....	55
Gambar 4.29 Perbandingan Hasil Pengujian web service Spring Boot dan Laravel Metode <i>POST</i> .....	56
Gambar 4.30 Perbandingan hasil pengujian web service Spring Boot dan Laravel metode <i>PUT</i> .....	57
Gambar 4.31 Perbandingan Hasil Pengujian Spring Boot dan Laravel metode <i>GET</i> .....	58
Gambar 4. 32 Perbandingan Hasil Pengujian Spring Boot dan Laravel dengan metode <i>DELETE</i> .....	59



## I PENDAHULUAN

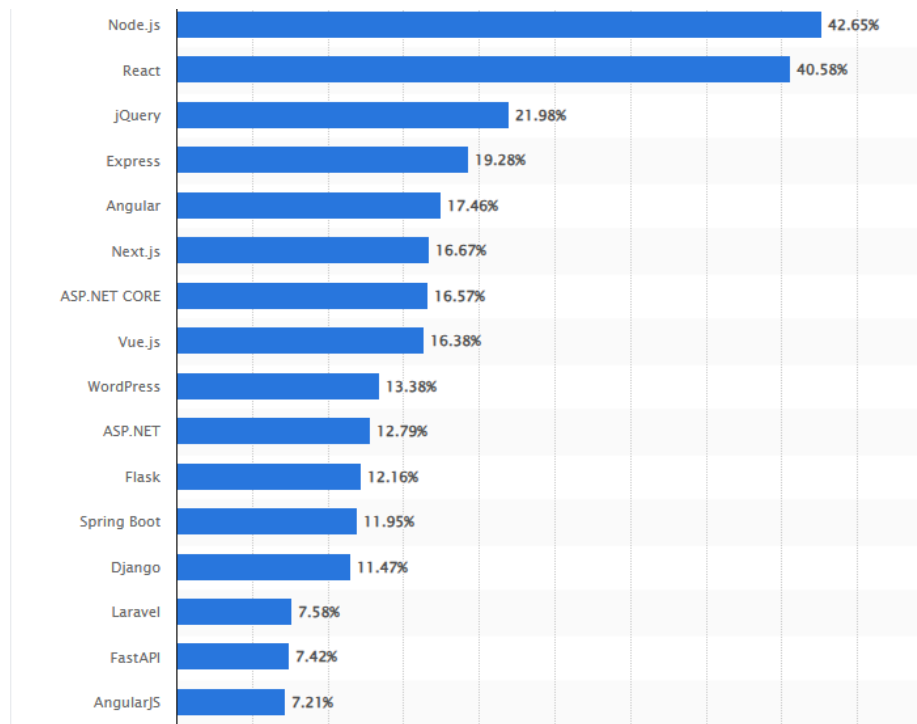
### 1.1 Latar Belakang

Penggunaan teknologi informasi dan komunikasi yang terus berkembang, memberikan pengaruh besar terhadap organisasi maupun individu dalam memenuhi kebutuhan informasi maupun komunikasi. Kemudahan dalam mengakses informasi antar perangkat dan media merupakan suatu fungsi yang sangat penting dalam pengembangan aplikasi untuk menjadi lebih besar, maka dari itu diperlukan integrasi sistem yang saling terhubung dalam berbagai macam *platform* serta perangkat yang ada dengan salah satu cara yang dapat dipilih adalah penggunaan *webservice*[1]

*Web service* adalah sebuah aplikasi perangkat lunak yang digunakan untuk para pengguna agar dapat melakukan komunikasi dengan perangkat keras sebagai sarana penyimpanan dan pengolahan informasi yang dibutuhkan. Dewasa ini *web service* sudah banyak digunakan untuk memfasilitasi berbagai kepentingan user yang diantaranya lainnya adalah untuk kebutuhan edukasi, komersial, dan pemerintahan[2]. Dengan kebutuhan tersebut sebuah *web service* perlu dibangun dengan cepat dan hal ini dapat dilakukan dengan menggunakan sebuah *framework*.

*Representational State Transfer Application Interface* (REST API) adalah sebuah cara bagaimana sebuah program klien dapat melakukan komunikasi dengan *web service* dimana REST adalah sebuah gaya arsitektur yang digunakan secara umum untuk *web service* modern dan API yang akan melakukan ekspose dari data dan fungsi yang dibutuhkan oleh program klien.

*Framework* atau kerangka kerja adalah sebuah alat untuk memudahkan para pembuat *web service* untuk membangun *web service* dengan cara yang lebih efisien dan lebih berkualitas dibandingkan dengan menggunakan bahasa pemrograman secara langsung untuk memproses kemudian mengolah dan menyimpan informasi dari para pengguna yang nantinya akan menggunakan *web service* tersebut. Hal ini dikarenakan sebuah *framework* itu sendiri terdiri atas kode kode yang sudah pernah ditulis oleh para pembuat *web service* lainnya yang dikumpulkan kemudian di rangkai sedemikian rupa sehingga kode tersebut dapat digunakan kembali oleh pembuat *web service* lain untuk menghemat waktu dan tenaga tanpa harus membuat sebuah *web service* dari awal. Untuk saat ini menurut website statista dalam 15 urutan *framework* yang paling terdapat *framework* Spring Boot yang menggunakan bahasa Java dan Laravel yang menggunakan bahasa PHP [3]. Spring boot adalah sebuah *framework* dari bahasa Java yang merupakan pecahan dari *framework* Spring yang berfokus khusus untuk membuat sebuah aplikasi berbasis web. Menurut website spring.io Spring Boot mempermudah pembuatan aplikasi berbasis Spring dengan membuat para pembuat *webservice* agar dapat langsung mengerjakan pembuatan aplikasi yang dengan kesulitan yang minimal [4]. Laravel adalah sebuah *framework* dari bahasa pemrograman Hypertext Processor (PHP) yang merupakan bahasa *scripting opensource* yang dikhususkan untuk pengembangan web yang muncul pada tahun 1995 dibuat oleh Rasmus Lerdorf[5]. Pada awalnya PHP adalah akronim dari Personal Homepage Tools. Laravel menggunakan sintaks yang elegan sehingga dapat membuat proses pengembangan menjadi luar biasa [6].



Gambar 1.1 Diagram batang 16 *framework* populer tahun 2023[3]

Dari gambar 1.1 diagram batang 16 *framework* yang populer digunakan pada tahun 2023 dapat dilihat bahwa Spring Boot dan Laravel masing masing menempati urutan nomor 12 dan 14. Kepopuleran ini didukung dengan analisis dari *website* Wappalyzer mengenai beberapa *website* terkenal yang menggunakan Spring Boot contohnya adalah aliexpress, alibaba, dan lazada[7], dan untuk *framework* Laravel digunakan oleh beberapa perusahaan seperti Twitch, Disney, The New York Times dan masih banyak lagi[6].

Kedua *framework* tersebut memiliki performa yang berbeda satu dengan yang lain sehingga dibutuhkan sebuah pengujian yang akan mengambil data-data dari kinerja web service yang dikembangkan menggunakan kedua *framework* tersebut. Untuk kebutuhan pengujian digunakan aplikasi yang bernama Apache Jmeter. Apache Jmeter ini adalah sebuah perangkat lunak pengujian yang berfokus pada pengujian aplikasi web[8]. Apache Jmeter ini memiliki kelebihan yaitu bersifat sumber terbuka dan memiliki fitur fitur yang cukup untuk melakukan pengujian sebuah aplikasi berbasis web. Menurut penelitian Rabiya Abbas dkk. yang melakukan perbandingan beberapa aplikasi pengujian didapatkan hasil yaitu Apache Jmeter ini memiliki kelebihan dibandingkan dengan alat pengujian lainnya. Selain itu Apache

Jmeter ini juga memiliki komunitas yang masih sangat aktif sehingga penambahan fitur baru dan perbaikan *bug* cukup sering dilakukan oleh para developer secara sukarela.

Penelitian ini akan melakukan proses rancang bangun 2 buah web service menggunakan *framework* Laravel dan Spring Boot yang masing masing web service akan di desain untuk melakukan proses pengolahan data *Create, Read, Update, dan Delete* (CRUD) yang nanti akan dikirimkan melalui aplikasi pengujian Apache Jmeter pada sebuah *database* yang sama. Hasil data pengujian kedua *web service* berupa waktu respon akan diambil dan dibandingkan dan diharapkan bisa di gunakan sebagai alasan mengapa pengembang melakukan pengembangan sebuah aplikasi web service dengan *framework* Laravel atau Spring Boot .

## **1.2 Rumusan Masalah**

Rumusan masalah dalam penelitian ini adalah bagaimana menguji performa waktu respon yang dapat dilayani tiap detik dari *web service* yang dibangun menggunakan *framework* Laravel dan Spring Boot

## **1.3 Tujuan Penelitian**

Tujuan dari penelitian ini adalah melakukan rancang bangun dua buah web service yang dibangun menggunakan *framework* Laravel dan Spring Boot, melakukan pengujian data performa *web service* yang dibangun untuk dianalisa perbedaan performanya.

## **1.4 Manfaat Penelitian**

Manfaat yang diperoleh dari pengujian ini adalah data yang didapatkan dari hasil pengujian dapat dijadikan perbandingan performa dari *framework* Laravel dan Spring Boot untuk pembuatan *web services*.

## 1.5 Batasan Masalah

Batasan masalah dilakukan agar penelitian lebih terarah dan memudahkan dalam pembahasan, sehingga tujuan dari penelitian dapat tercapai. Batasan masalah dalam pengembangan sistem ini yaitu:

1. Pengujian dilakukan untuk mengetahui performa dari masing masing *framework* untuk melakukan kegiatan *Create, Read, Update, dan Delete* (CRUD) dalam sebuah *database*.
2. Pengujian dilakukan secara lokal dan *database* yg digunakan adalah XAMPP.
3. *Database* yang digunakan adalah *database* relasional MariaDB yang sama.
4. Pengujian dilakukan tanpa memperhatikan masalah keamanan dari masing masing *framework*.

## 1.6 Sistematika Penulisan

Sistematika penulisan yang digunakan pada laporan penelitian ini adalah sebagai berikut:

### **BAB I : PENDAHULUAN**

Bab ini menguraikan secara umum mengenai latar belakang penelitian, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan penelitian, dan sistematika penulisan.

### **BAB II : TINJAUAN PUSTAKA**

Bab ini berisi tentang prinsip, landasan teori, dan penelitian yang terkait sebagai sumber dalam memahami penelitian perbandingan performa api web service menggunakan *framework* seperti Comparative study on Python web frameworks: Flask and Django, A Comparative Investigation on Implementation of RESTful versus SOAP based Web Services, A Performance Comparison of RESTful Applications Implemented in Spring Boot Java and MS.NET Core, Analisis Komparasi Performa Web Application: Studi Kasus Asp.Net

Mvc Dan Asp.Net Core, Analisis Perbandingan Performa Web Service Rest Menggunakan Framework Laravel, Django, Dan Node Js Pada Aplikasi Berbasis Website

**BAB III : METODE PENELITIAN**

Bab ini membahas mengenai metode penelitian yang digunakan dalam perbandingan performa api *web service* menggunakan *framework* Spring Boot dan laravel

**BAB IV : HASIL DAN PEMBAHASAN**

Bab ini berisi tentang pembahasan serta hasil yang diperoleh dalam penelitian.

**BAB V : KESIMPULAN DAN SARAN**

Bab ini berisi kesimpulan dari hasil penelitian dan saran-saran sebagai masukan untuk penelitian lanjutan di masa mendatang.

**DAFTAR PUSTAKA**

**LAMPIRAN**

## II. TINJAUAN PUSTAKA

### 2.1 *Web Service*

Menurut W3C (World Wide Web Consortium) *Web Service* Adalah sebuah sistem perangkat lunak yang didesain untuk mendukung pengoprasian interaksi dari mesin ke mesin melalui sebuah jaringan dan memiliki antarmuka yang dapat di deskripsikan dalam format yang dimengerti oleh mesin. Sistem lain yang ingin melakukan interaksi dengan *Web Service* biasanya menggunakan *Hypertext Transfer Protocol* (HTTP) atau *Extensible Markup Language* (XML)[9]

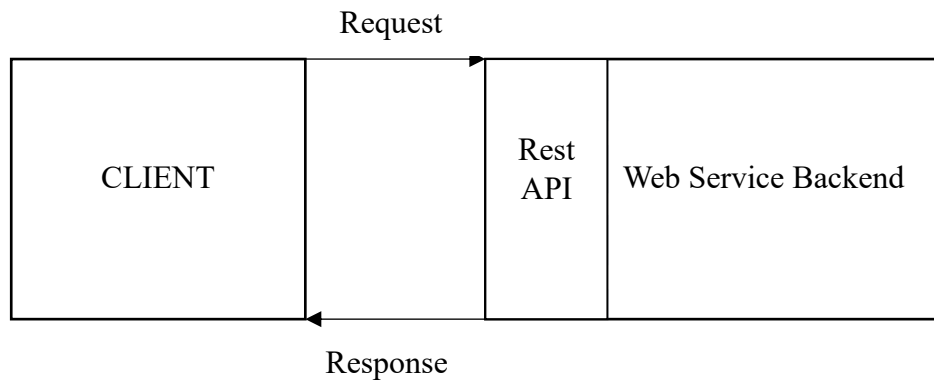
### 2.2 **Hyper Text Transfer Protocol (HTTP)**

HTTP adalah sebuah protokol standar yang digunakan dalam internet. HTTP ini memiliki unsur yang ringan dan cepat yang dibutuhkan oleh sistem informasi kolaboratif *hypermedia* yang terdistribusi. HTTP ini memiliki ciri berupa protokol berorientasi objek yang tidak memiliki *state* dan berisikan koneksi dimana klien akan melakukan kepada server, *Request* adalah pengiriman permintaan sebuah informasi kepada server oleh klien, *Response* adalah sebuah jawaban dari server mengenai Request yang dikirimkan oleh klien, *Close* adalah pemutusan koneksi dari kedua pihak. Dalam sebuah *request* yang dikirimkan klien terdapat metode yang akan dilakukan oleh server pada objek yang diminta dalam *request* diantaranya adalah metode GET yang memiliki fungsi untuk mengambil data yang diminta oleh klien, PUT memiliki fungsi untuk mengubah data di dalam *server* yang diminta dengan data yang dikirimkan oleh klien, DELETE adalah metode yang digunakan untuk meminta server menghapus data yang diminta oleh klien, POST adalah metode yan digunakan untuk membuah objek baru dalam server [10].

### 2.3 **Rest API**

Menurut buku REST API Design RuleBook *API* (Aplication Programming Interface) adalah sebuah cara dari *client* untuk melakukan komunikasi dengan sebuah *Web Service*. API mengekspose kumpulan data data dan fungsi untuk

memfasilitasi interaksi antara program komputer dan membolehkan mereka untuk melakukan pertukaran informasi[11]. Hal ini diilustrasikan dalam dari gambar 2.1



Gambar 2.1 Diagram Rest API

Dari gambar 2.1 dapat dilihat *Client* akan melakukan *Request* ke sebuah *Web Service* akan diterima oleh dahulu di Rest API sebelum diproses oleh *Web Service Backend* nya begitu pula sebaliknya untuk mengirim *Response* yang di client maka *Web Service* akan mengirimnya melalui API sebelum menuju ke *Client*.

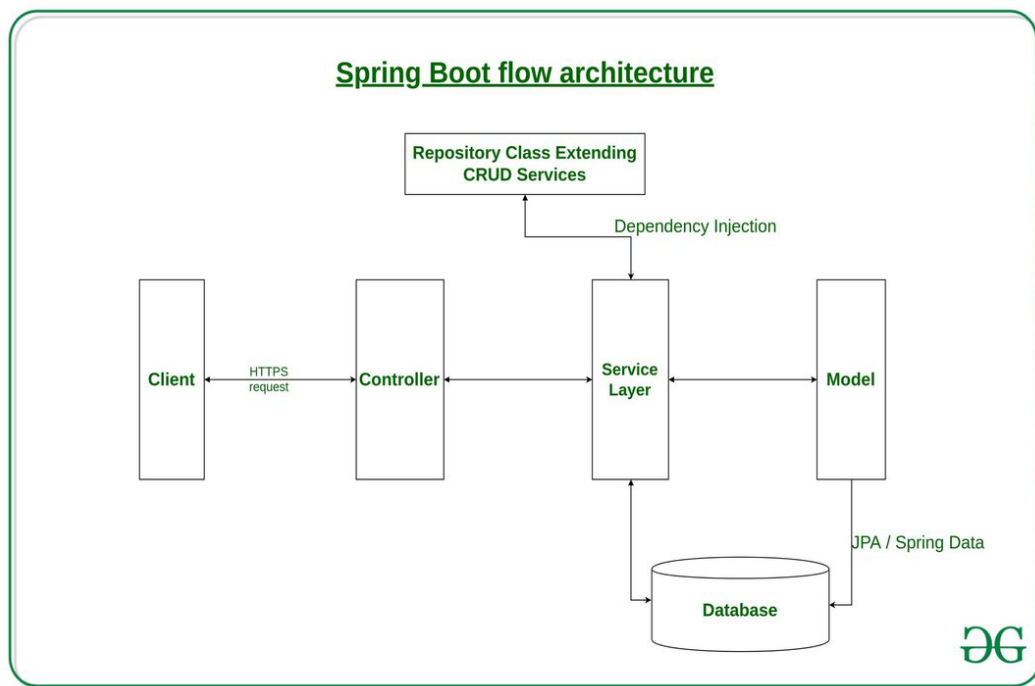
#### **2.4 Framework**

*Framework* adalah sebuah aplikasi semi komplit yang bisa diberikan sebuah spesialisasi untuk membuat sebuah aplikasi kustom dan dapat di gunakan kembali untuk mengembangkan aplikasi yang lain. Pengembang aplikasi desktop konvensional akan mengharapkan aplikasi memiliki kesamaan dalam cara penerimaan, penyimpanan dan pengolahan data. *Framework* aplikasi dibangun karena hal tersebut untuk menyediakan para pengembang aplikasi sebuah struktur yang dapat digunakan berkali kali yang bisa menjadi pondasi untuk produk mereka. *Framework* memberikan para pengembang kumpulan dari komponen kerangka yang memiliki beberapa karakteristik yaitu dapat bekerja dengan baik dengan aplikasi lain, framework siap untuk digunakan dalam proyek selanjutnya, dan framework dapat dimengerti dan digunakan oleh tim lain dalam organisasi[11].



## 2.5 Spring Boot

Spring boot adalah sebuah *framework* atau kerangka kerja yang menggunakan bahasa pemrograman Java. Spring boot adalah sebuah pecahan dari framework Java yang memiliki nama sama yaitu Spring dengan kelebihan yaitu Spring boot mempermudah semua pengembang dalam melakukan proses pengembangan dengan meringkas proses konfigurasi *library* pihak ketiga yang akan ditambahkan kedalam aplikasi yang sedang dikembangkan. Konfigurasi penambahan ini dapat dilakukan oleh *framework* Spring Boot dengan mudah dan juga Spring Boot dapat secara otomatis melakukan pengecekan terhadap versi dari *library* tambahan ini sehingga terjadinya konflik versi yang tidak sesuai dapat dihindari dan mencegah hambatan produktivitas pengembang [12].

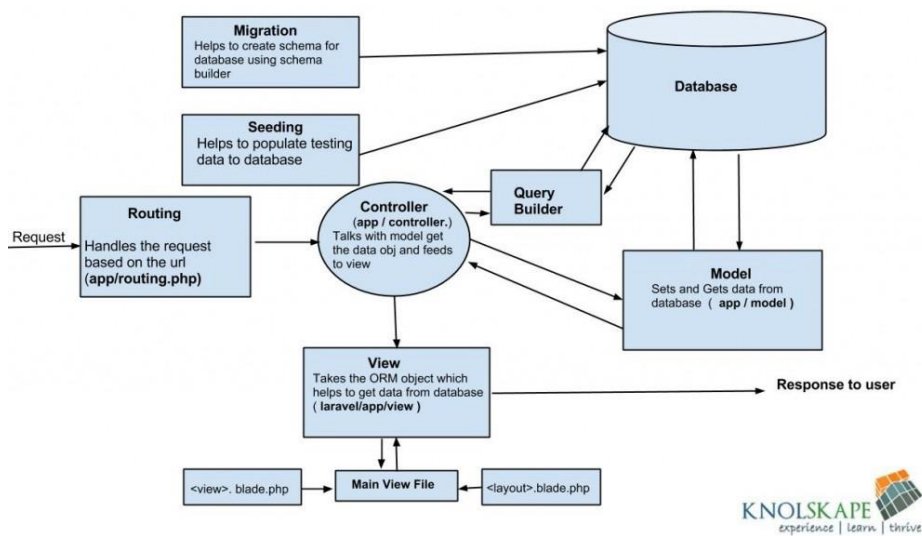


Gambar 2.2 Diagram *framework* Spring Boot[13]

## 2.6 Laravel

Adalah sebuah *Framework* yang menggunakan bahasa PHP yang keluar pada tahun 2011. Laravel adalah sebuah *framework* pengembangan aplikasi cepat yang artinya adalah Laravel melakukan fokus terhadap kurva pembelajaran yang mudah dan

meminimalisir langkah langkah yang diperlukan untuk memulai aplikasi baru dan melakukan publikasi dari aplikasi yang sudah dibuat. Semua kegiatan yang lumrah pada pembangunan aplikasi web dari *database*, interaksi, autentikasi, antrian email sampai *caching* dibuat mudah oleh laravel. Laravel juga memberikan ekosistem untuk alat alat pembangunan dan publikasi *website* selain itu Laravel ini memiliki komunitas yang ramah dan mau membantu sehingga membuat *Framework* Laravel ini dapat berkembang seperti sekarang[14]. Laravel membuat pengembangan sistem aplikasi berbasis web menjadi lebih terstandarisasi dan dapat melakukan proses relasi antara logis non bisnis secara otomatis[15].



Gambar 2.3 Diagram *Framework* Laravel

## 2.7 Pengujian performa

Pengujian performa adalah salah satu bagian dari tahapan Software Development LifeCycle (SDLC). Pengujian ini dilakukan untuk menentukan apakah sebuah sistem telah memenuhi kebutuhan non fungsional yang terdapat di dokumen Software Requirement Specification (SRS) atau tidak. Pengujian performa ini dapat dilakukan dengan cara mengukur waktu respon dan *throughput* dari sebuah aplikasi web[16]. Pengujian yang dilakukan adalah pengujian *load test* dimana sebuah aplikasi akan diuji dengan memberikan muatan beban penggunaan dan kemudian mencatat perilaku dari sistem terhadap beban yang diterima[17].

## 2.8 Aggregate Report jmeter

*Aggregate Report* adalah sebuah tabel yang mengukur performa waktu respon dari sebuah *Web Service*[17]. Untuk melakukan pengukuran performa ini akan digunakan sebuah aplikasi yang bernama Apache Jmeter. Apache Jmeter memiliki sebuah fungsi yang bernama *listener* yang dapat melakukan pengawasan terhadap matriks performa untuk mengukur waktu respons dan persen dari error. Waktu respons di definisikan sebagai waktu total yang dibutuhkan untuk sebuah *web service* untuk melayani sebuah *request* yang dibuat oleh klien kepada sebuah *web service* dan diukur dalam satuan *milisecond*. Dan persen kegagalan adalah sebuah matriks pengukuran yang mengukur tingkat persentase dari *request* yang gagal dalam melakukan koneksi dengan *web service*. Dalam aplikasi Jmeter akan dibuatkan sebuah pengguna virtual yang akan melakukan *request* kepada *web service* yang sudah dibuat dan akan dilakukan monitor terhadap *request* yang dilakukan dan hasilnya akan ditampilkan.



Gambar 2.4 Logo Apache Jmeter

## 2.9 XAMPP

XAMPP adalah sebuah aplikasi yang berbasis *opensource* yang digunakan untuk melakukan instalasi sebuah Apache Web Server yang kemudian dapat digunakan untuk menjalankan aplikasi web dalam lingkungan lokal. Selain itu menurut website resmi PHP XAMPP ini juga memiliki fitur aplikasi tambahan penunjang dalam proses pembuatan aplikasi berbasis web seperti MariaDB, PHP dan Perl[18].



Gambar 2.5 Logo XAMPP

## **2.10 Penelitian terkait**

Penelitian terkait merupakan penelitian yang telah dilakukan sebelumnya. Diharapkan penelitian yang telah dilakukan sebelumnya menjadi sebuah pendukung dalam melakukan penelitian ini. Berikut merupakan beberapa penelitian yang dijadikan referensi dalam penelitian ini:

### **2.10.1 Analisis Perbandingan Performa Web Service Rest Menggunakan Framework Laravel, Django, Dan Node Js Pada Aplikasi Berbasis Website**

Pada penelitian yang dilakukan oleh Abi Amrulloh dkk[1]. Dapat dilihat bahwa penelitian ini membandingkan tiga *framework web service* yaitu Nodejs, Laravel dan Django. Dalam metode pengujiannya pertama dibuat sebuah database dengan beberapa tabel yang nantinya akan dipopulasikan kemudian langkah selanjutnya adalah membuat API Web Service yang akan disambungkan dengan database yang sudah dibuat tadi dengan ketiga *Framework* yang sudah disebutkan dan langkah terakhir adalah dengan melakukan pengujian dengan aplikasi Apache Benchmark untuk melakukan request GET pada 3 Web Service yang dibuat untuk mengakses data yang tersimpan dalam database. Dari hasil akhir penelitian didapatkan bahwa Laravel adalah Framework yang paling banyak *request* yang gagal dengan 3940 *request* untuk kedua framework lainnya 0. Untuk *framework* yang mempunyai kecepatan yang paling tinggi tiap *request* adalah NodeJS dengan total waktu 182.014 detik untuk menyelesaikan pengujian dan untuk framework yang memiliki jumlah paling tinggi untuk perbandingan *request* tiap detik adalah Django dengan 11.60 *request* tiap detik

### **2.10.2 Analisis Komparasi Performa Web Application: Studi Kasus Asp.Net Mvc Dan Asp.Net Core**

Menurut penelitian yang dilakukan oleh Enry Christanto dan Tony Wibowo[19] mereka melakukan perbandingan dari dua framework yaitu ASP.NET MVC dan ASP.NET CORE. kedua *framework* ini dibuat oleh perusahaan Microsoft perbedaan dari framework ini adalah ASP.NET MVC menggunakan paradigma MVC atau Model, View dan Controller sedangkan untuk ASP.NET Core adalah sebuah desain

ulang dari dari ASP.NET yang dapat digunakan untuk membangun hal yang lebih luas secara *cross platform*. Dalam penelitian ini dilakukan pengujian kedua *framework* dalam segi performa, konsumsi daya memori dengan menerima dan memproses data. Dari penelitian didapatkan hasil yaitu ASP.NET MVC memiliki waktu respon yang lebih sedikit daripada ASP.NET CORE tetapi ASP.NET CORE menggunakan sumberdaya yang lebih sedikit dalam penggunaannya.

### **2.10.3 A Performance Comparison of RESTful Applications Implemented in Spring Boot Java and MS.NET Core**

Dari penelitian yang dilakukan Hardeep Kaur Dhalla [2] yang melakukan perbandingan antara framework Spring Boot Java dan MS.NET Core. Metode yang digunakan dalam penelitian ini adalah dengan menguji kedua Framework dengan teknik Create Update Read Delete dengan data buatan dan kemudian menggunakan aplikasi Jmeter untuk melakukan proses Request untuk masing masing Web Service dan memonitor hasil yang didapatkan. Hasil yang didapatkan dari pengujian ini adalah bahwa MS.NET CORE memiliki waktu respon yang lebih singkat dan memakan sumberdaya memori yang lebih sedikit dibandingkan framework Spring Boot

### **2.10.4 A Comparative Investigation on Implementation of RESTful versus SOAP based Web Services**

Dalam penelitian yang dilakukan oleh Abhijit Bora dan Tulshi Bezboruah[20] yang melakukan investigasi komparasi dalam implementasi RESTful dan SOAP ini bertujuan untuk menilai protokol mana yang lebih superior dalam pembuatan aplikasi *Web Service*. Pada penelitian ini dibangun sebuah *web service* dengan protokol SOAP da RESTful menggunakan bahasa JAVA yang memiliki fungsi sebagai *Web Service* pencarian obat dan penyakit. Dari pengujian didapatkan hasil berupa protokol RESTful adalah yang lebih bagus daripada protokol SOAP.

### **2.10.5 Comparative study on Python web frameworks: Flask and Django**

Dari penelitian yang dilakukan oleh Devndra Ghimire [21] yang bertujuan untuk mempelajari kelebihan dan kekurangan serta batasan dari fitur fitur yang dimiliki oleh *framework* Python yaitu Flask dan Django. Dalam penelitian dibuat sebuah *web service* yang berbentuk lokapasar menggunakan kedua *framework* yang

terhubung kedalam database dan di *deploy* menggunakan *platform* Heroku. Hasil dari penelitian ini adalah Flask lebih memberikan fleksibilitas dan kemudahan dalam proses pembelajaran daripada *framework* Django.

### 2.10.6 Comparative Study of Load Testing Tools: Apache JMeter, HP LoadRunner, Microsoft Visual Studio (TFS), Siege

Dari penelitian yang dilakukan oleh Rabbiya Abbas dkk[22] yang memiliki tujuan untuk mengkomparasikan alat alat *load testing* yaitu Apache Jmeter, HP Load Runner, Microsoft Visual Studio(TFS). Software tersebut itu akan digunakan untuk melakukan pengujian *website* <http://www.buic.edu.pk/>, dan [www.telenor.pk](http://www.telenor.pk). Dari pengujian didapatkan bahwa aplikasi pengujian Apache Jmeter memiliki hasil yang lebih baik daripada aplikasi pengujian lainnya.

Tabel 2.1 Tabel ringkasan penelitian

No	Nama Pengarang	Judul Penelitian	Metode	Hasil
1.	Abi Amarulloh, Kurniasih, Muchlis(2023 )	ANALISIS PERBANDINGAN PERFORMA WEB SERVICE REST MENGGUNAKAN FRAMEWORK LARAVEL, DJANGO, DAN Node JS PADA APLIKASI BERBASIS WEBSITE	<ul style="list-style-type: none"> <li>- Framework yang digunakan adalah NodeJs, Laravel dan Django</li> <li>- Menggunakan aplikasi Apache Benchmark</li> <li>- Melakukan method GET pada tiga <i>framework</i> yang terhubung pada database sama</li> </ul>	Laravel memiliki nilai request gagal paling banyak yaitu 3940 <i>request</i> , NodeJs memiliki waktu penyelesaian tercepat yaitu 182.014 detik dan Django memiliki kemampuan <i>request</i> tiap detik paling banyak yaitu 11.60 <i>request</i> tiap detik
2.	Enry Christanto,	ANALISIS KOMPARASI PERFORMA	- <i>Framework</i> yang digunakan adalah	ASP.NET CORE mendapatka

	Tony Wibowo(2020)	WEB APPLICATION: STUDI KASUS ASP.NET MVC DAN ASP.NET CORE	<p>ASP.NET MVC dan ASP.NET CORE</p> <ul style="list-style-type: none"> <li>- Menggunakan database yang sama untuk kedua <i>framework</i></li> <li>- Metode pengujian dilakukan dengan 2 komputer yaitu host untuk menjalankan aplikasi Jmeter dan remote untuk melakukan hosting API yang telah dibuat</li> <li>- Metode yang dilakukan hanya GET untuk mengambil data dalam database dengan jumlah request 500 sampai 10000</li> </ul>	n waktu respon yang lebih lambat dibandingkan ASP.NET MVC
3.	Hardeep Kaur Dhalla(2021)	A Performance Comparison of RESTful Applications Implemented in Spring Boot Java and MS.NET Core	<ul style="list-style-type: none"> <li>- Framework yang digunakan adalah Spring boot dan MS&gt;NET Core</li> <li>- Metode pengujian menggunakan aplikasi Jmeter untuk memberikan <i>request</i> CRUD kepada api masing masing <i>framework</i></li> </ul>	MS.NET CORE memiliki waktu respon yang lebih singkat dan menggunakan sumberdaya memori yang lebih kecil daripada Springboot
4.	Devndra Ghimire(2021)	A Comparative Investigation on Implementation of RESTful versus SOAP based Web Services	<ul style="list-style-type: none"> <li>- Perbandingan untuk desain web service yaitu SOAP dan RESTful</li> <li>- Kedua desain web service dibangun menggunakan bahasa JAVA dan database Mysql</li> <li>- Menggunakan-an Mercury Load Runner sebagai aplikasi pengujian</li> </ul>	Restful lebih cepat dibandingkan dengan SOAP

5.	Abhijit Bora, Tulshi Bezboruah(2015)	Comparative study on Python web frameworks: Flask and Django	<ul style="list-style-type: none"> <li>- Framework yang digunakan adalah python Flask dan Django</li> <li>- Kedua framework dibuatkan web service berupa lokapasar yang terhubung database MySQL</li> <li>- melakukan hosting kedua web service dengan heroku</li> </ul>	Flask lebih mudah untuk dipelajari lebih simpel dan fleksibel, sedangkan Django memiliki fitur yang lebih banyak dan lebih mudah untuk di <i>scaling</i>
6	Rabiya Abbas, Zainab Sultan, Shahid Nazir Bhatti, Farrukh Latif Butt(2017)	Comparative Study of Load Testing Tools: Apache JMeter, HP LoadRunner, Microsoft Visual Studio (TFS), Siege	<ul style="list-style-type: none"> <li>- melakukan komparasi aplikasi pengujian pada website <a href="http://www.buic.edu.pk/">http://www.buic.edu.pk/</a>, <a href="http://www.buic.edu.pk/">http://www.buic.edu.pk/</a></li> <li>- aplikasi yang digunakan adalah Apache JMeter, HP LoadRunner, Microsoft Visual Studio (TFS), Siege</li> </ul>	Aapache Jmeter memiliki hasil yang lebih baik daripada aplikasi pengujian lainnya

Dari penelitian-penelitian terkait yang terdapat pada tabel 2.1 mempunyai kesamaan yaitu pengujian dari *web service* dengan sebuah *framework* tertentu dengan menggunakan aplikasi Jmeter. Sehingga dalam skripsi ini akan dilakukan pengujian antara dua framework yaitu Laravel dan Spring Boot ini yaitu dalam hal melakukan perbandingan performa *web service* yang dibuat dengan *framework* Laravel dan *framework* Spring Boot selanjutnya akan diuji menggunakan aplikasi *benchmark* Jmeter.



### III. METODOLOGI PENELITIAN

#### 3.1 Waktu dan Tempat

Penelitian ini dimulai pada semester genap tahun ajaran 2023 dengan tempat pelaksanaan di Program Studi Teknik Informatika, Jurusan Teknik Elektro, Fakultas Teknik, Universitas Lampung. Berikut ini pada tabel 3.1 merupakan alur penelitian:

Tabel 3.1 Jadwal Penelitian

No	Aktivitas	Mei 2023	Juni 2023	Juli 2023	Agustus 2023	September 2023
1	<i>Literature Review</i>					
2	Perancangan Database					
3	Perancangan Web Service					
4	Pengujian dengan Jmeter					
5	Penarikan Kesimpulan dan Penulisan Laporan					

## 3.2 Alat Dan Bahan

### 3.2.1 Alat dan bahan penelitian

Adapun alat dan bahan yang digunakan dalam penelitian ini adalah sebagai berikut

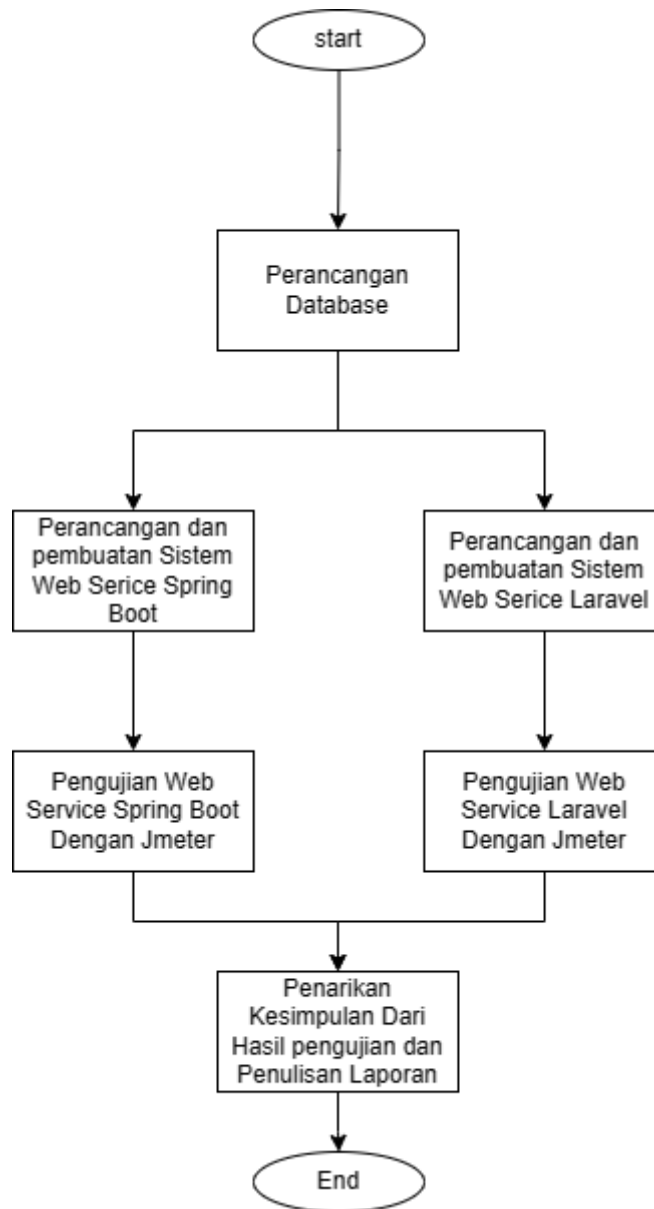
Tabel 2.2 Alat dan bahan penelitian

No	Nama Alat	Spesifikasi	Deskripsi
1	Laptop Asus X441UB	Intel® Core™ i3-6006U dengan RAM 8GB dan sistem operasi Windows 10	Perangkat keras yang digunakan untuk melakukan proses pengembangan aplikasi
2	Intelij Idea	Versi 2023.1	Perangkat lunak yang digunakan untuk pengembangan aplikasi berbahasa Java
3	Visual Studio Code	Versi 1.81.1	Perangkat lunak untuk melakukan pengembangan aplikasi Laravel
4	Java	Versi 18.0.1.1	Bahasa pemrograman untuk mengembangkan aplikasi dengan framework Spring Boot
5	Springboot	Versi 3.0.1	Framework Java untuk mengembangkan aplikasi berbasis Web
6	Apache Maven	Versi 3.8.6	Perangkat lunak untuk manajemen proyek Java
7	PHP	Versi 8.2.10	Bahasa pemrograman yang digunakan untuk mengembangkan aplikasi framework dengan Laravel

8	Laravel	Versi 10.1.7	Framework PHP untuk mengembangkan aplikasi Web
9	Apache Jmeter	Versi 5.6.1	Aplikasi untuk melakukan pengujian <i>web service</i>
10	MariaDB	Versi 10.4.28	Database untuk menyimpan data yang dikirimkan.

### **3.3 Tahapan Penelitian**

Pada penelitian ini digunakan metode kuantitatif untuk mengukur perbandingan performa antara dua aplikasi *web service* yang akan dibangun. Kedua aplikasi ini akan diuji berdasarkan dari waktu respon yang diukur dengan aplikasi Jmeter.



Gambar 3.1 Alur Penelitian

Dari gambar 3.1 Alur Penelitian dapat dilihat bahwa pengujian akan diawali dengan melakukan perancangan *database* yang menggunakan *database* MariaDB pada aplikasi xampp kemudian dilanjutkan dengan rancang bangun dua buah *web service* menggunakan framework Spring Boot dan Laravel dengan fungsionalitas sama yaitu untuk melakukan operasi pengolahan data kedalam *database* yang sudah dirancang pada awal proses penelitian. Operasi pengolahan yang dilakukan adalah Create digunakan untuk memasukkan data, Read digunakan untuk membaca data, Update digunakan untuk mengubah data, dan Delete digunakan untuk menghapus

data. Setelah rancang bangun maka akan dilakukan pengujian beban menggunakan aplikasi Apache Jmeter untuk mengirimkan *payload* berisi data yang akan dikirimkan ke web service yang sudah dijalankan menggunakan *server* bawaan masing masing framework. *Payload* akan dikirimkan oleh aplikasi Apache Jmeter menggunakan protokol HTTP *request* dengan metode Post untuk menguji metode Create, Get untuk menguji metode read, Put untuk menguji metode update, dan Delete untuk menguji metode delete. Performa dari ke dua web service dalam menjalankan pengujian ini yang berupa waktu respon akan dicatat dan selanjutnya akan dibandingkan antara satu dengan yang lain untuk kemudian akan ditarik kesimpulan framework mana yang lebih unggul.

### 1. Perancangan Database

Database digunakan untuk menyimpan informasi yang akan disimpan dan di akses oleh pengguna melalui API yang akan dibuat. Rancangan *database* adalah seperti gambar 3.2

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	id	bigint(20)		No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/>	2	alamat	varchar(255)	utf8mb4_general_ci	Yes	NULL			Change  Drop  More
<input type="checkbox"/>	3	email	varchar(255)	utf8mb4_general_ci	Yes	NULL			Change  Drop  More
<input type="checkbox"/>	4	nama	varchar(255)	utf8mb4_general_ci	Yes	NULL			Change  Drop  More
<input type="checkbox"/>	5	no_hp	varchar(255)	utf8mb4_general_ci	Yes	NULL			Change  Drop  More
<input type="checkbox"/>	6	password	varchar(255)	utf8mb4_general_ci	Yes	NULL			Change  Drop  More

Gambar 3.2 Tabel Database Pengujian

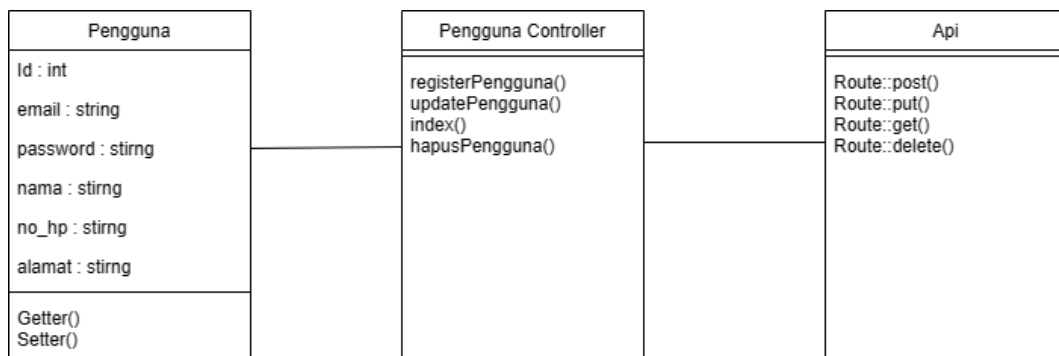
Dari tabel *database* akan mempunyai beberapa *field* yaitu id dengan tipe *integer* sebagai *Primary key*, alamat dengan tipe *varchar*, email dengan tipe *varchar*, nama dengan tipe *varchar*, no\_hp dengan tipe *varchar*, dan yang terakhir *password* dengan tipe *varchar*.

Dalam penerapannya pada kedua *framework* baik Spring Boot dan Laravel akan dihubungkan dengan *database* MariaDB yang sama sebagai penyimpanan data. Kedua framework ini mempunyai sebuah fitur bawaan yang bernama ORM (*Object Relational Mapping*). ORM ini akan melakukan representasi dari tabel yang ada di dalam *database* sebagai sebuah objek yang dapat dikenali oleh masing masing

*framework* untuk dapat melakukan operasi dari perintah perintah CRUD yang akan dibuat dan digunakan dalam pengujian.

## 2. Perancangan *Web Service*

Pada tahap ini akan dirancang dua buah *web service* dengan masing masing *framework* Spring Boot dan Laravel dengan fungsionalitas yang sama yaitu pembuatan sebuah API dengan 4 *endpoint Controller POST, PUT, GET,DELETE* yang masing masing memiliki fungsi *Create, Update Read, dan Delete* untuk memanipulasi data yang terdapat dalam database yang sudah dirancang.



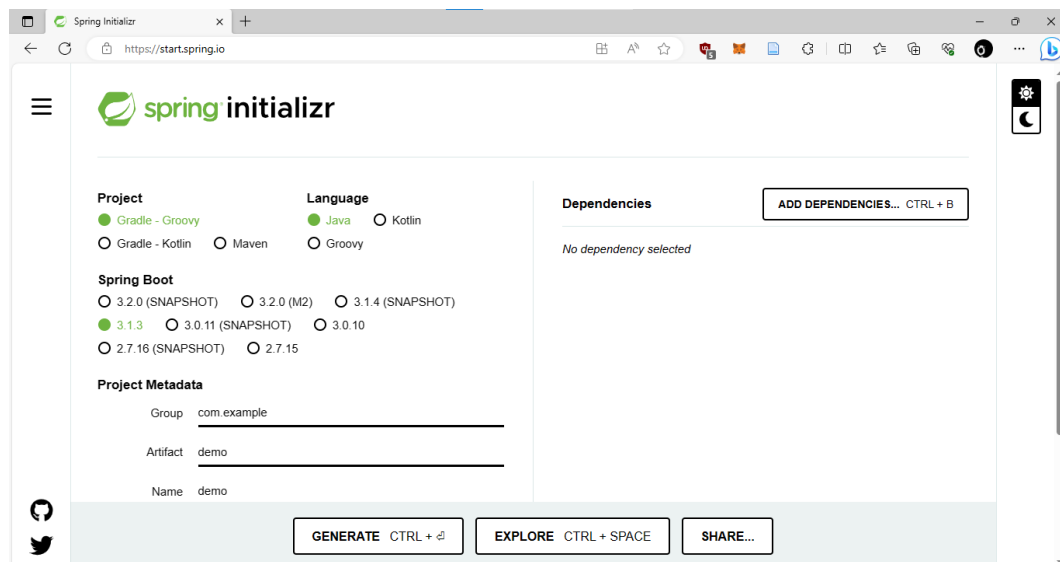
Gambar 3.3 class diagram web service

Dari gambar 3.3 dapat dilihat *class diagram* yang nantinya akan dibuat pada masing masing *web service*. Terdapat 3 class yaitu class Pengguna sebagai model yang akan menjadi objek penyimpanan data pada database yang sudah di olah *web service* pada class Pengguna Controller. Pengguna Controller digunakan untuk melakukan pengolahan data dengan 4 fungsi yaitu registerPengguna untuk menambahkan data, updatePengguna untuk melakukan perubahan data, index() digunakan untuk membaca seluruh data, dan hapusPengguna digunakan untuk menghapus data. fungsi fungsi ini dapat digunakan dengan cara mengakses alamat pada class API. Class API akan menyimpan rute yang akan digunakan untuk mengakses fungsi fungsi yang terdapat pada class penggunaController rute yang terdapat pada class ini adalah route::post() yang akan digunakan untuk mengakses fungsi registerPengguna, route::put() digunakan untuk mengakses fungsi updatePengguna, route::get() digunakan untuk mengakses fungsi index, dan route::delete() digunakan untuk mengakses fungsi hapusPengguna.

Kedua web service akan dijalankan menggunakan server bawaan yang sudah terdapat pada waktu proses inisiasi dari masing masing project web service. Untuk web service Spring Boot server bawaan yang digunakan adalah Tomcat dan untuk Laravel server yang digunakan adalah server bawaan dari bahasa program PHP itu sendiri.

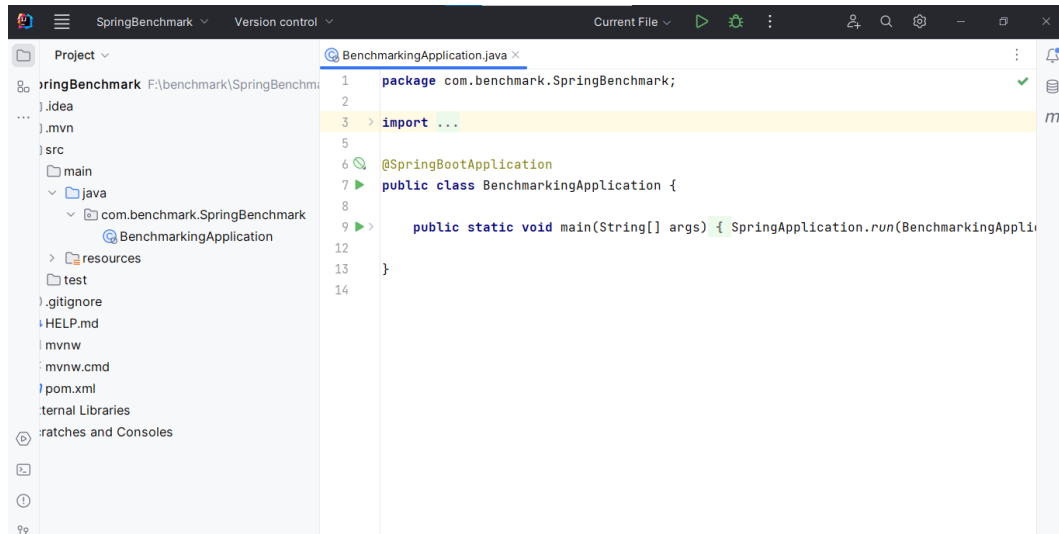
## 2.1 Perancangan *Framework* Spring Boot

Hal yang perlu dilakukan untuk pertama kali dalam proses perancangan aplikasi *web service* menggunakan *framework* Spring Boot adalah dengan inisiasi dari project yang akan dibuat. Hal ini dilakukan dengan mengakses halaman spring initializr pada url <https://start.spring.io/>.



Gambar 3.4 Halaman Muka Spring Initializr

Pada gambar 3.4 Dapat dilihat halaman muka dari web Spring Initializr. Dalam halaman ini terdapat pilihan pilihan yang dapat digunakan untuk melakukan konfigurasi *web service* yang akan dibuat seperti memilih jenis proyek, bahasa yang digunakan, versi dari Spring Boot yang akan digunakan, metadata dari proyek, dan *Dependencies* yang akan digunakan. *Dependencies* adalah *library* dengan fungsi tambahan yang dibutuhkan dalam aplikasi yang akan dibuat. Jika semua konfigurasi sudah selesai maka klik *Generate* untuk melakukan unduh kerangka proyek yang sudah dibuat.



Gambar 3.5 Proyek yang sudah di ekstrak

Setelah selesai diunduh dan dilakukan proses ekstraksi kemudian dapat dibuka menggunakan IDE IntelliJ Idea. Saat dibuka menggunakan IntelliJ Idea maka akan tampak tampilan dari struktur proyek seperti yang terdapat pada gambar 3.5 untuk melakukan proses coding.

## 2.2 Perancangan *Framework* Laravel

Perancangan menggunakan *framework* Laravel diawali dengan pembuatan *project* menggunakan sebuah aplikasi composer yang dapat diunduh pada link berikut <https://getcomposer.org/download/>.



```
Command Prompt
Microsoft Windows [Version 10.0.19045.3324]
(c) Microsoft Corporation. All rights reserved.

C:\Users\User>composer

Composer version 2.6.2 2023-09-03 14:09:15

Usage:
  command [options] [arguments]

Options:
  -h, --help                Display help for the given command. When no command is given display help for the list
  command
  -q, --quiet               Do not output any message
  -V, --version              Display this application version
  --ansi|--no-ansi          Force (or disable --no-ansi) ANSI output
  -n, --no-interaction      Do not ask any interactive question
  --profile                 Display timing and memory usage information
  --no-plugins              Whether to disable plugins.
  --no-scripts              Skips the execution of all scripts defined in composer.json file.
  -d, --working-dir=WORKING-DIR If specified, use the given directory as working directory.
  --no-cache                Prevent use of the cache
  -v|vv|vVV, --verbose     Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and
  3 for debug
```

Gambar 3.6 Composer berhasil di instalasi

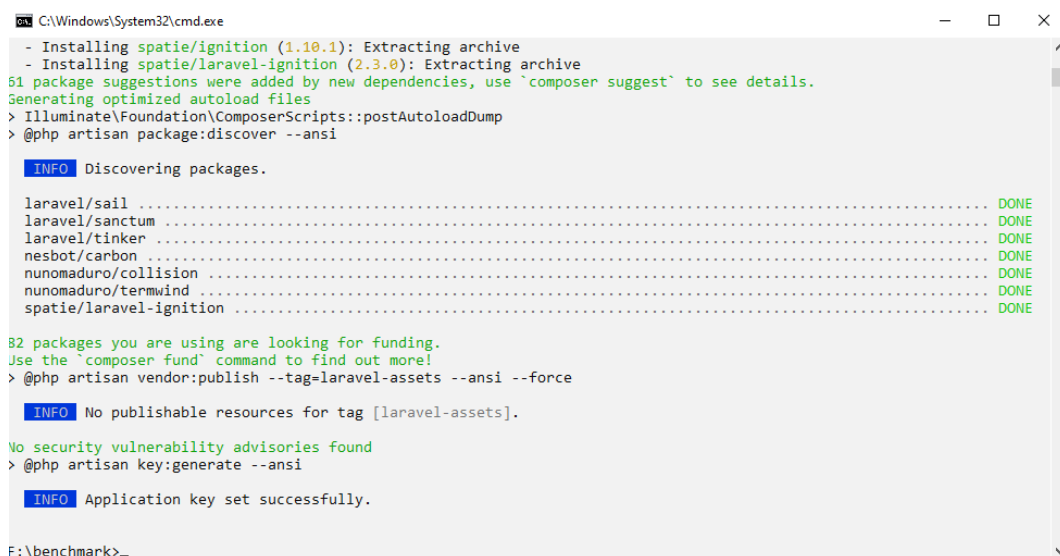
Jika sudah berhasil dilakukan instalasi maka saat mengetikkan composer di Command Prompt akan mengeluarkan pilihan pilihan yang terdapat pada gambar 3.6 setelah terkonfirmasi sudah berhasil maka barulah dapat dilakukan untuk memasukkan perintah inisiasi proyek *framework* Laravel.

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.3324]
(c) Microsoft Corporation. All rights reserved.

F:\benchmark>composer create-project laravel/laravel LaravelBenchmark
Creating a "laravel/laravel" project at "./LaravelBenchmark"
Info from https://repo.packagist.org: #StandWithUkraine
Installing laravel/laravel (v10.2.6)
 - Installing laravel/laravel (v10.2.6): Extracting archive
Created project in F:\benchmark\LaravelBenchmark
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
Updating dependencies
Lock file operations: 110 installs, 0 updates, 0 removals
 - Locking brick/math (0.11.0)
 - Locking dflydev/dot-access-data (v3.0.2)
 - Locking doctrine/inflexor (2.0.8)
 - Locking doctrine/lexer (3.0.0)
 - Locking dragonmantank/cron-expression (v3.3.3)
 - Locking egulias/email-validator (4.0.1)
 - Locking fakerphp/faker (v1.23.0)
 - Locking filp/whoops (2.15.3)
 - Locking fruitcake/php-cors (v1.2.0)
 - Locking graham-campbell/result-type (v1.1.1)
 - Locking guzzlehttp/guzzle (7.8.0)
 - Locking guzzlehttp/promises (2.0.1)
 - Locking guzzlehttp/psr7 (2.6.1)
 - Locking guzzlehttp/uri-template (v1.0.2)
 - Locking hamcrest/hamcrest-php (v2.0.1)
 - Locking laravel/framework (v10.23.0)
 - Locking laravel/pint (v1.13.1)
```

Gambar 3.7 perintah untuk membuat proyek framework laravel dengan nama LaravelBenchmark

Untuk melakukan inisiasi sebuah proyek *framework* laravel dengan nama `LaravelBenchmark` dibutuhkan perintah “`composer create-project laravel/laravel LaravelBenchmark`” dan menekan enter. Setelah ditekan enter maka composer akan melakukan pengunduhan *dependencies* bawaan dari laravel yang dibutuhkan dalam sebuah proyek *framework* laravel. Berbeda dengan Spring Boot pengunduhan *dependencies* ini akan mengunduh banyak *package* yang belum tentu digunakan dalam sebuah aplikasi *web service* yang akan dikembangkan sehingga struktur file dari *project* akan lebih rumit dibandingkan dengan struktur file project Spring Boot.



```
C:\Windows\System32\cmd.exe
- Installing spatie/ignition (1.10.1): Extracting archive
- Installing spatie/laravel-ignition (2.3.0): Extracting archive
51 package suggestions were added by new dependencies, use `composer suggest` to see details.
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi

 INFO  Discovering packages.

laravel/sail ..... DONE
laravel/sanctum ..... DONE
laravel/tinker ..... DONE
nesbot/carbon ..... DONE
nunomaduro/collision ..... DONE
nunomaduro/termwind ..... DONE
spatie/laravel-ignition ..... DONE

82 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
> @php artisan vendor:publish --tag=laravel-assets --ansi --force

 INFO  No publishable resources for tag [laravel-assets].

No security vulnerability advisories found
> @php artisan key:generate --ansi

 INFO  Application key set successfully.

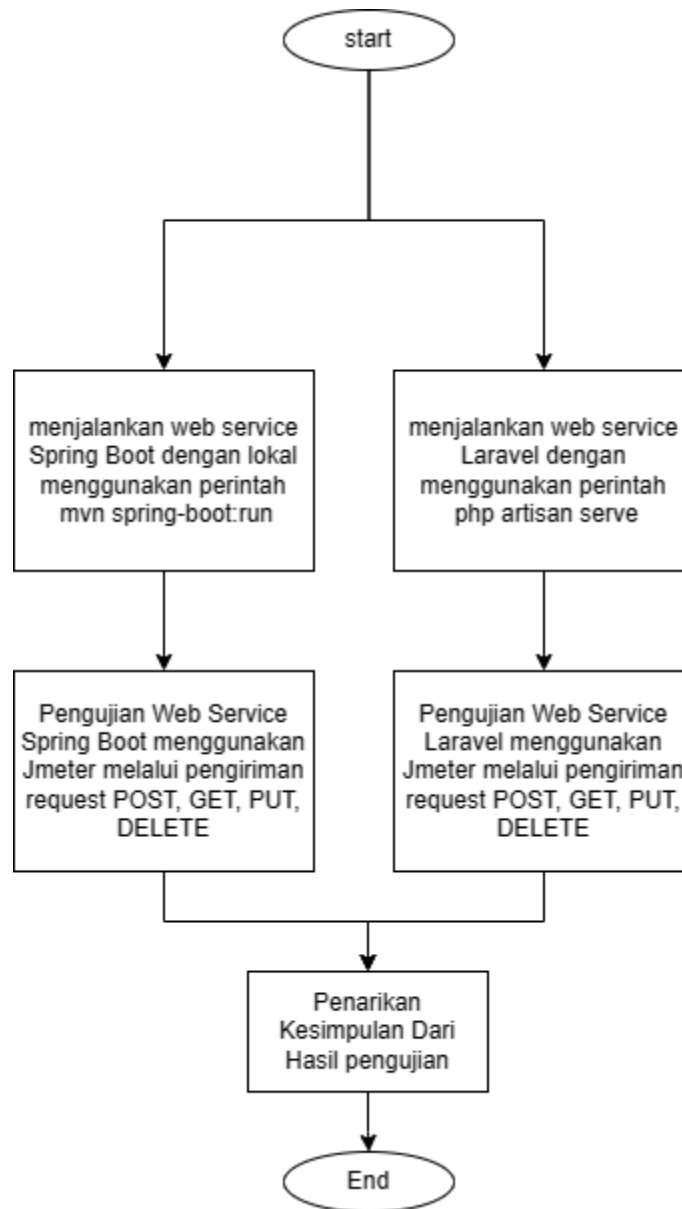
F:\benchmark>
```

Gambar 3.8 *output* saat instalasi berhasil

Jika instalasi berhasil maka akan terdapat keluaran seperti yang terdapat pada gambar 3.8 didalam *command prompt* yang menunjukkan pesan bahwa *dependencies* yang diunduh oleh komposer sukses ditambahkan dalam proyek `LaravelBenchmark`.

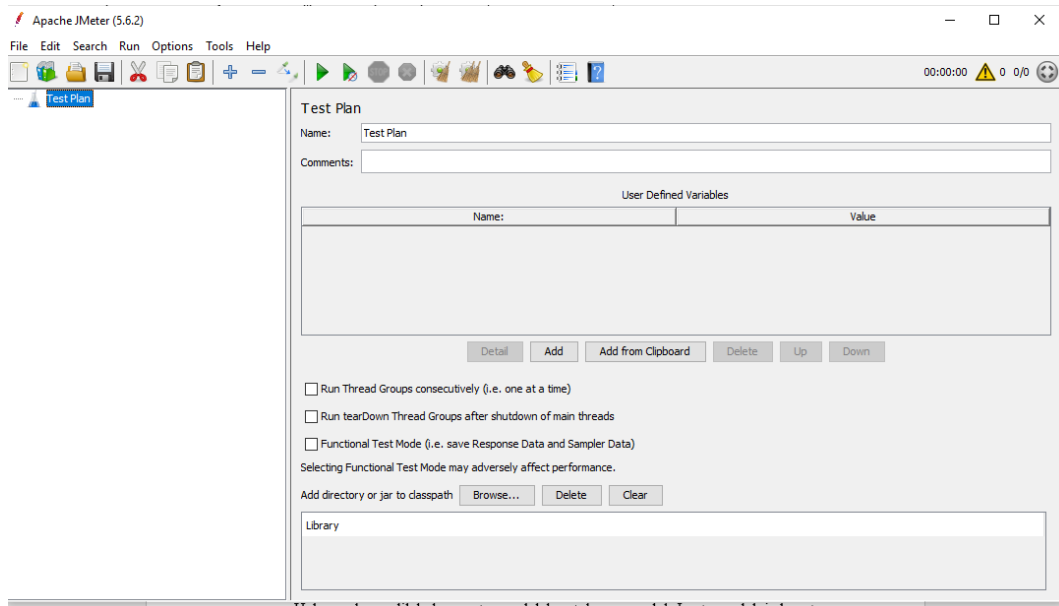
### 3. Pengujian dengan Jmeter

Pada tahap ini *web service* yang sudah dirancang akan dijalankan dalam *environment* lokal kemudian akan dilakukan pengujian menggunakan aplikasi Apache Jmeter dengan jumlah request 10000 untuk masing masing *endpoint* dari 4 controller API yang dimiliki oleh masing *web service*.



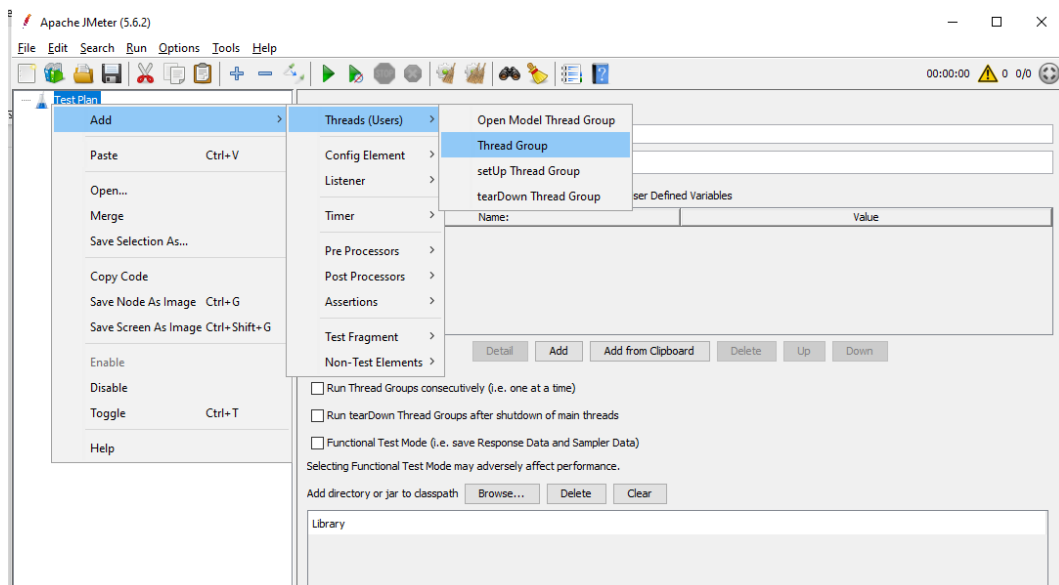
Gambar 3.9 Rancangan Pengujian

Dari gambar 3.9 dapat dilihat untuk tahapan pengujian diawali dengan menjalankan aplikasi web service yang sudah dibuat secara lokal dengan mengetikkan perintah `mvn spring-boot:run` untuk aplikasi web service spring boot, dan perintah `php artisan serve` untuk aplikasi web service Laravel. Kedua perintah ini akan menjalankan server bawaan yang dimiliki oleh masing-masing framework sehingga web service dapat diakses dan dilakukan pengujian. Masing-masing server bawaan yang dijalankan menggunakan setelan konfigurasi bawaan. Pengujian antara *web service* dilakukan secara terpisah agar hasil yang didapatkan tidak dipengaruhi oleh konsumsi dari sumberdaya yang terbagi.



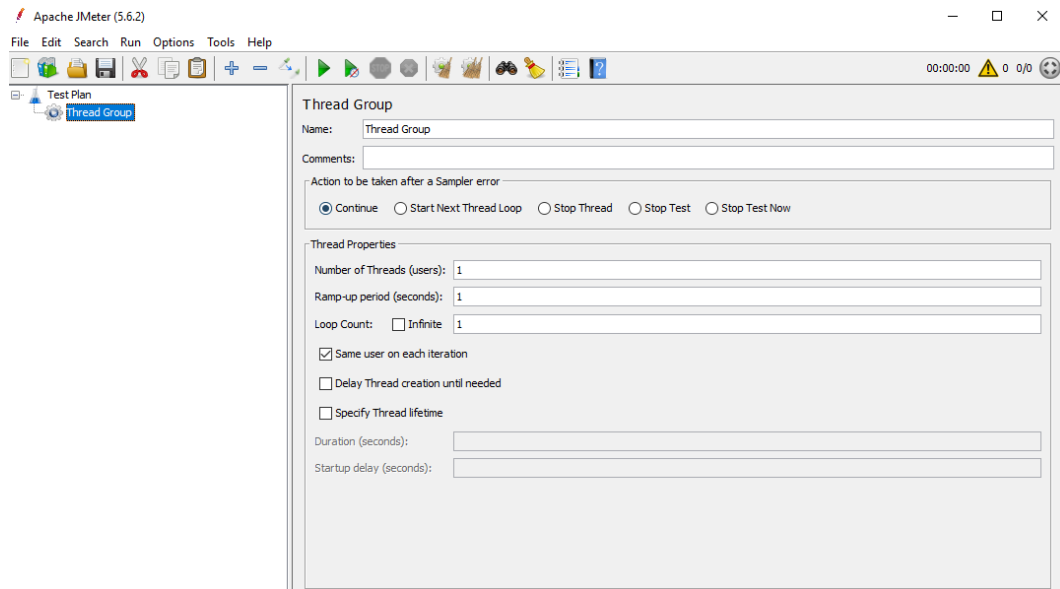
Gambar 3.10 Tampilan antarmuka aplikasi Jmeter

Hal yang harus dilakukan pertama adalah untuk mengunduh Jmeter melalui alamat [https://d1cdn.apache.org//jmeter/source/apache-jmeter-5.6.2\\_src.zip](https://d1cdn.apache.org//jmeter/source/apache-jmeter-5.6.2_src.zip) kemudian melakukan ekstraksi dari file yang sudah diunduh. Selanjutnya masuk kedalam folder hasil ekstraksi dan menuju kedalam folder bin. Setelah itu buka file yang bernama jmeter.bat untuk membuka program jmeter dengan versi yang memiliki tampilan grafis.



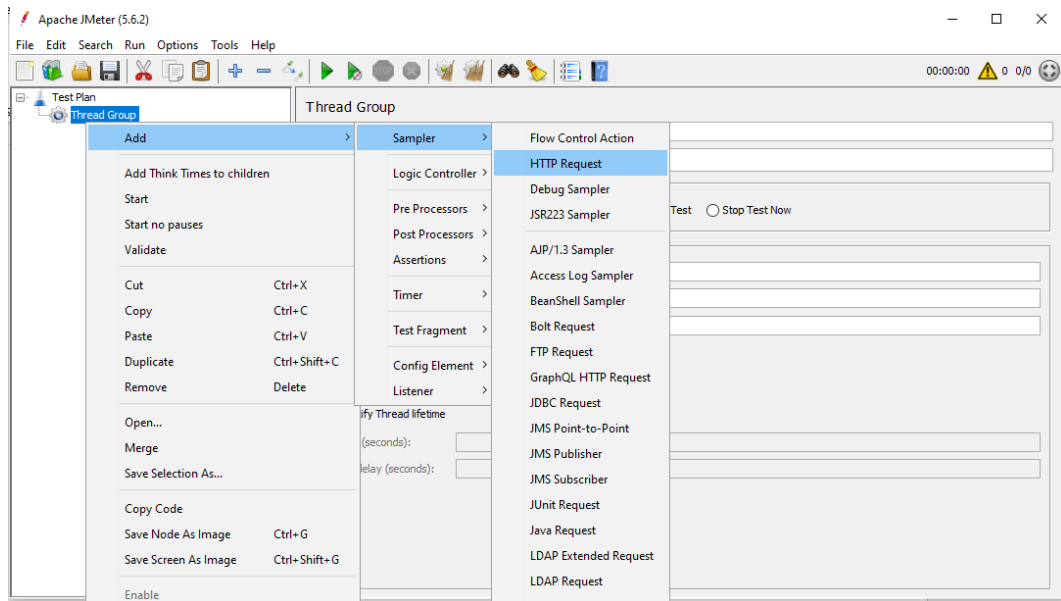
Gambar 3. 11 Penambahan *Thread Group*

Langkah selanjutnya adalah untuk membuat *virtual user* yang dapat ditemukan dalam menu test plan dan memilih pilihan Add Threads(User) dan ThreadGroups dapat dilihat dari gambar 3.11. Ini akan menjadi grup *virtual user* yang akan melakukan request kepada web service yang sudah dibuat.



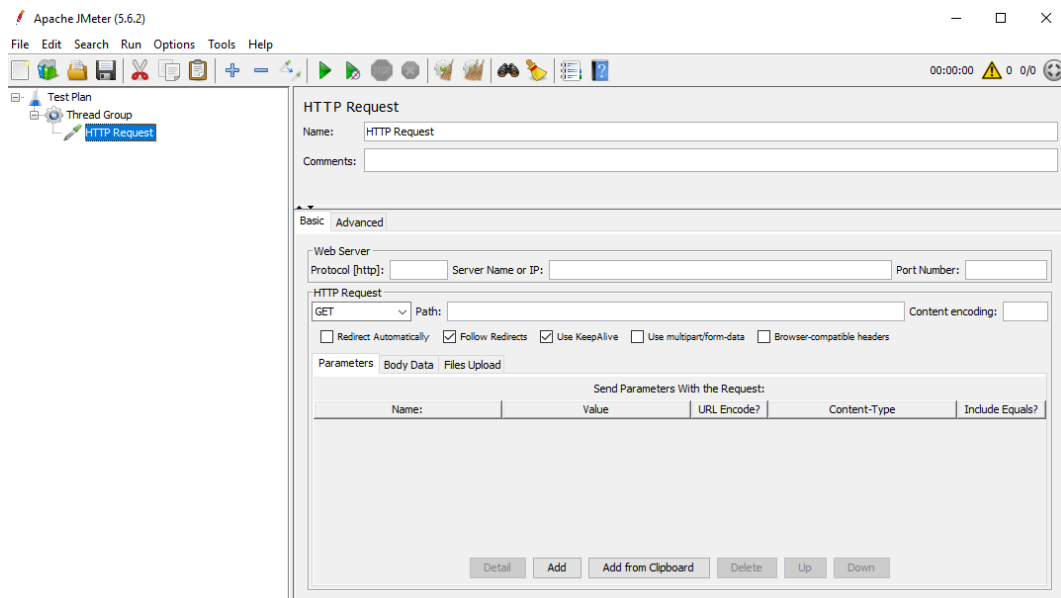
Gambar 3.12 *Virtual User*

Dalam gambar 3.12 dapat dilihat proses yang dilakukan untuk menentukan jumlah *virtual user* yang akan digunakan dalam proses pengujian pada kotak *Number Of Threads(User)* dan delay dari setiap *request* yang dikirimkan dengan *Ramp-Up Periods* jika dimasukkan 0 maka permintaan akan dilakukan dengan instan dan seolah olah terjadi dalam waktu bersamaan.



Gambar 3.13 Menambahkan *http request*

Kemudian untuk menambahkan *request* yang akan dilakukan untuk proses pengujian dapat dilakukan dengan cara memilih menu dropdown pada Thread group memilih opsi add kemudian sampler dan terakhir opsi HTTP Request. Untuk lebih jelasnya dapat dilihat pada gambar 3.13 Menambahkan *http request*



Gambar 3.14 *Http Request*

Setelah dibuat maka akan muncul tampilan gambar 3.14. disini terdapat opsi untuk mengubah parameter isi dari *request* yang akan dikirimkan ke *web service* seperti alamat API dari *web service* yang akan dilakukan pengujian dan *method* yang akan diujikan seperti *Post,Get,Update* dan *Delete*. Jika sudah memasukkan parameter yang diinginkan maka dapat memulai pengujian dengan lambang arah segitiga warna hijau dibagian atas.

#### **4. Penarikan Kesimpulan dan Penulisan Laporan**

Pada tahap ini hasil yang didapatkan dari tahap pengujian dengan aplikasi Apache Jmeter akan dikumpulkan dan akan dilakukan penghitungan rata-rata yang didapatkan oleh masing masing *web service* setelah didapatkan barulah akan diambil kesimpulan mengenai hasil uji coba yang didapatkan untuk dituliskan kedalam laporan akhir.

## V. KESIMPULAN

### 5.1 Kesimpulan

Kesimpulan yang diperoleh berdasarkan hasil dari penelitian adalah sebagai berikut

1. Dari pengujian yang dilakukan diambil jumlah respon dan waktu respon dari kedua *web service* yang dibangun menggunakan Spring Boot dan Laravel. Didapatkan bahwa pada 10000 *request* dengan metode *Post, Put, Get*, dan *Delete* berturut-turut Spring Boot memiliki keunggulan rata rata waktu respon yaitu 44,8ms, 45,37ms, 109,5ms, 42,ms dibandingkan Laravel yaitu 291,98ms, 301,2ms, 426,54ms, 273,06ms. Keunggulan ini juga ditunjukkan pada pengujian menggunakan 1000, 3000, dan 5000 *request* yang bisa dilihat pada lampiran.
2. Keunggulan *web service* yang dibangun dengan Spring Boot terjadi dikarenakan *web service framework* Spring Boot secara *default* melakukan unduhan *dependencies* yang lebih sedikit dibandingkan dengan *dependencies* yang diunduh secara *default* pada *framework* Laravel.
3. Masing-masing *framework* dapat menyelesaikan seluruh *request* pengujian dengan baik. Hal ini dibuktikan dengan tingkat *error* kedua *framework* memiliki nilai 0% untuk tiap metode CRUD.
4. Berdasarkan data yang didapatkan pada pengujian metode *GET Web Service* Spring Boot memiliki keunggulan selain waktu rata-rata yang lebih cepat Spring Boot juga dapat mengirimkan data lebih banyak dengan 11567.18 KB data tiap detik.



## 5.2 Saran

Saran yang dapat diberikan untuk melanjutkan penelitian ini berdasarkan hasil dari penelitian ini adalah sebagai berikut .

1. Untuk pengujian selanjutnya dapat dilakukan proses *deploy* kepada sebuah server yang dedicated atau melakukan proses kontainerisasi menggunakan docker untuk tiap web service.
2. Melakukan pemantauan penggunaan resource server oleh masing masing web service
3. Menambah sebuah skenario yang menggunakan kasus pada dunia nyata dalam pembuatan *web service* seperti melakukan registrasi, login, logout dan menambahkan data seperti postingan.

## Daftar Pustaka

- [1] “Analisis Perbandingan Performa Web Service Rest Menggunakan Framework Laravel, Django, Dan Node Js Untuk Akses Data Dengan Aplikasi Website | Jurnal Teknik Informatika.” Accessed: Aug. 23, 2023. [Online]. Available: <https://ejournal.antarbangsa.ac.id/jti/article/view/515>
- [2] H. K. Dhalla, “A Performance Comparison of RESTful Applications Implemented in Spring Boot Java and MS.NET Core,” *J. Phys. Conf. Ser.*, vol. 1933, no. 1, p. 012041, Jun. 2021, doi: 10.1088/1742-6596/1933/1/012041.
- [3] “Most used web frameworks among developers 2023,” Statista. Accessed: Aug. 29, 2023. [Online]. Available: <https://www.statista.com/statistics/1124699/worldwide-developer-survey-most-used-frameworks-web/>
- [4] “Spring Boot,” Spring Boot. Accessed: Aug. 29, 2023. [Online]. Available: <https://spring.io/projects/spring-boot>
- [5] “Announce: Personal Home Page Tools (PHP Tools).” Accessed: Aug. 29, 2023. [Online]. Available: <https://groups.google.com/g/comp.infosystems.www.authoring.cgi/c/PyJ25gZ6z7A/m/M9FkTUVDFcwJ?pli=1>
- [6] “Laravel - The PHP Framework For Web Artisans.” Accessed: Aug. 29, 2023. [Online]. Available: <https://laravel.com/>
- [7] “Websites using Spring - Wappalyzer.” Accessed: Sep. 27, 2023. [Online]. Available: <https://www.wappalyzer.com/technologies/web-frameworks/spring/>
- [8] “Apache JMeter - Apache JMeter™.” Accessed: Oct. 30, 2023. [Online]. Available: <https://jmeter.apache.org/index.html>
- [9] “Web Services at W3C: Web Service.” Accessed: Aug. 31, 2023. [Online]. Available: <https://www.w3.org/2005/Talks/01-ws-plh/?n=1>
- [10] “HTTP: A protocol for networked information.” Accessed: Nov. 30, 2023. [Online]. Available: <https://www.w3.org/Protocols/HTTP/HTTP2.html>
- [11] M. H. Massé and M. Massé, *REST API design rulebook: designing consistent RESTful Web Service Interfaces*. Beijing Köln: O’Reilly, 2012.
- [12] T. Husted and C. Dumoulin, *Struts in action: building web applications with the leading Java framework*. Greenwich, CT: Manning, 2003.
- [13] C. Walls, *Spring Boot in action*. Shelter Island, NY: Manning Publications, 2016.
- [14] “Spring Boot - Architecture,” GeeksforGeeks. Accessed: Sep. 26, 2023. [Online]. Available: <https://www.geeksforgeeks.org/spring-boot-architecture/>
- [15] M. Stauffer, “Laravel: Up and Running: A Framework for Building Modern PHP Apps. Gravenstein Highway North, Sebastopol,: O’Reilly, 2016”.
- [16] R. Y. He, “design and implementation of web based on laravel framework:,” presented at the 2014 International Conference on Computer Science and Electronic Technology (ICCSET 2014), ShenZhen, China, 2015. doi: 10.2991/iccset-14.2015.66.
- [17] S. Dhiman and P. Sharma, “Performance Testing: A Comparative Study and Analysis of Web Service Testing Tools,” 2016.

- [18] S. Matam and J. Jain, *Pro Apache JMeter*. Berkeley, CA: Apress, 2017. doi: 10.1007/978-1-4842-2961-3.
- [19] "About the XAMPP project." Accessed: Sep. 25, 2023. [Online]. Available: <https://www.apachefriends.org/about.html>
- [20] E. Christanto and T. Wibowo, "Analisis Komparasi Performa Web Application: Studi Kasus Asp.Net Mvc Dan Asp.Net Core," vol. 1, 2020.
- [21] A. Bora and T. Bezboruah, "A Comparative Investigation on Implementation of RESTful versus SOAP based Web Services," *Int. J. Database Theory Appl.*, vol. 8, no. 3, pp. 297–312, Jun. 2015, doi: 10.14257/ijdta.2015.8.3.26.
- [22] D. Ghimire, "Comparative study on Python web frameworks: Flask and Django, B.S. thesis, Media Eng., Metropolia University of Applied Sciences., Finland, 2020".
- [23] R. Abbas, Z. Sultan, and S. N. Bhatti, "Comparative Study of Load Testing Tools: Apache JMeter, HP LoadRunner, Microsoft Visual Studio (TFS), Siege," *Sukkur IBA J. Comput. Math. Sci.*, vol. 1, no. 2, pp. 102–108, Dec. 2017, doi: 10.30537/sjcms.v1i2.24.