

Implementasi *Machine Learning Model Deployment* Pada Website Pemantauan Kondisi Sungai Citarum Menggunakan *Platform-As-A-Service*

1st Pietha Tiara Lingga Dewi
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia
pipiethatiara@student.telkomuni-
versity.ac.id

2nd Favian Dewanta
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia
favian@telkomuniversity.ac.id

3rd Muhammad Arief Nugroho
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia
arief.nugroho@telkomuniversity.
ac.id

Abstrak—*Machine learning* merupakan salah satu teknologi yang melakukan pembelajaran pada mesin berdasarkan data. *Machine learning* memiliki alur kerja yang kompleks yaitu *training* dan *development*. Pada tahap *development*, permasalahan ini dapat diselesaikan dengan pemanfaatan *cloud computing*. Salah satu layanan *cloud* yang dapat digunakan untuk membantu tahap pengembangan ini yaitu *Platform-as-a-Service*. *Platform-as-a-Service* merupakan salah satu layanan *cloud* yang sudah dilengkapi dengan fitur *hardware*, *software*, dan infrastruktur yang dapat memudahkan *user* dalam melakukan *deployment*. Tugas Akhir kali ini membuat sebuah website yang memiliki fitur prediksi berdasarkan data yang sudah diolah menggunakan *machine learning* dan membentuk sebuah model dan akan dilakukan pengujian terhadap performansi dari kedua layanan *Platform-as-a-Service* yang akan digunakan yaitu Heroku dan Microsoft Azure. Parameter pengujian yaitu *throughput*, *latency*, *load time*, *CPU usage*, dan *memory usage*. Pengujian akan dilakukan menggunakan *software Apache Jmeter*. Hasil akhir yang di dapatkan yaitu *cloud server* yang layak digunakan sebagai *platform deployment model* adalah Heroku dengan jumlah *user* 300 menghasilkan *throughput* 23.6 kbps dengan error 0.00%, *latency* sebesar 2305 ms, dan *load time* sebesar 2.5 second.

Kata kunci— *model machine learning*, *cloud server*, *Platform-as-a-Service*, *Deployment*

Abstract—*Machine learning* is a machine technology that performs data-based learning. *Machine learning* has a complex workflow, namely *training* and *development*. At the *development* stage, this problem can be solved by using *cloud computing*. One of the *cloud services* that can be used to assist in this *development* stage is *Platform-as-a-Service*. *Platform-as-a-Service* is a *cloud service* that is equipped with *hardware*,

software, and *infrastructure* features that can make it easier for users to deploy. In this final test, a website will be formed that has a prediction feature based on data that has been processed using machine learning and formed a model and will be tested on the performance of the two *Platform-as-a-Service* services that will be used, namely Heroku and Microsoft Azure. . The test parameters are *throughput*, *latency*, *load time*, *CPU usage*, and *memory usage*. Testing will be carried out using *Apache Jmeter* software. The final result that is obtained is that a *cloud server* that is suitable for use as a *model deployment platform* is Heroku with 300 users resulting in a *throughput* of 23.6 kbps with an error of 0.00%, *latency* of 2305 ms, and *load time* of 2.5 seconds.

Keywords — *model machine learning*, *cloud server*, *Platform-as-a-Service*, *Deployment*

I. PENDAHULUAN

Indonesia merupakan salah satu negara yang memiliki iklim tropis dengan dua musim yaitu musim kemarau dan hujan. Indonesia juga sering mengalami beberapa bencana alam. Salah satu bencana alam yang sering terjadi dan berkaitan dengan cuaca di Indonesia adalah banjir. Peristiwa ini sering melanda beberapa daerah salah satu contohnya yaitu Kabupaten Bandung, terkhususnya di Kecamatan Dayeuhkolot. Berdasarkan survey lingkungan yang dilakukan penyebab banjir dikarenakan adanya kenaikan intensitas air pada Sungai Citarum sehingga air tidak dapat ditampung oleh dinding sungai. Selain itu, daerah tersebut juga terletak di cekungan Kabupaten Bandung sehingga menyebabkan permukaan Sungai Citarum lebih tinggi dibandingkan

permukaan darat daerah Kecamatan Dayeuhkolot. Bencana ini sulit diantisipasi karena kurangnya informasi mengenai kondisi dari sungai tersebut. Berdasarkan dari permasalahan tersebut mulai dikembangkannya berbagai macam teknologi. Salah satu teknologi tersebut yaitu implementasi *machine learning* untuk membantu masyarakat dalam memprediksi kondisi dari Sungai Citarum.

Pembelajaran *machine learning* merupakan salah satu inovasi teknologi yang dapat membantu *developer* untuk mengelola suatu data dalam jumlah yang besar dengan waktu yang singkat [1]. Alur kerja sebuah *machine learning* sangat kompleks dan tersebar luas di berbagai domain. Ada dua tahap penting yang ada di pengembangan *machine learning* yaitu tahap *training* dan tahap *deployment*. Hal ini yang menyebabkan *machine learning* menjadi salah satu metode pengolahan data yang terkemuka. Namun, karena berkembangnya teknologi hal ini mendorong pertumbuhan skala pada beban kerja *machine learning* dan semakin meningkatkan kompleksitas dari alur kerja *machine learning* [2]. Dalam melakukan tahap pengembangan *deployment*, *machine learning* perlu memanfaatkan *server* komputasi jarak jauh atau dapat disebut dengan *cloud computing*. Infrastruktur untuk penyimpanan dan analitik data yang besar dapat melakukan pemrosesan lebih lanjut dan perlu dikembangkan secara efisien dan hemat biaya. Hal ini dapat diatasi melalui pemanfaatan *cloud computing* [3].

Arsitektur *cloud* mengacu pada berbagai komponen untuk membentuk suatu sistem awan. Arsitektur ini terdiri dari fitur *database*, perangkat lunak, dan aplikasi yang digunakan untuk membantu kinerja sumber daya *cloud* dalam memecahkan berbagai masalah. Di dalam arsitektur ini terdapat beberapa elemen dasar yaitu infrastruktur, penyimpanan, aplikasi, manajemen, dan keamanan [4]. Ada beberapa model layanan yang dimiliki oleh *cloud* yaitu *Infrastructure-as-a-Service*, *Software-as-a-Service*, dan *Platform-as-a-Service*. *Infrastructure-as-a-Service* merupakan salah satu layanan yang mengizinkan pengguna untuk menggunakan perangkat keras dan lunak. Namun, pengguna tidak benar-benar memiliki akses atas kendali perangkat keras tersebut [5]. *Software-as-a-Service* merupakan layanan yang

dirancang dari pihak sistem sendiri. Hal itu menyebabkan *Software-as-a-Service* tidak memberikan izin penuh kepada user untuk mengganti fitur-fitur yang sudah disediakan [6]. *Platform-as-a-Service* merupakan layanan pengembangan aplikasi berbasis web yang menyediakan *end-to-end* dalam beberapa kasus pengembangan. *Platform-as-a-Service* sudah terdapat beberapa fitur yaitu keamanan, *platform*, *Operation System*, *database*, *web server*, dan *framework* aplikasi sendiri [7].

Pada Tugas Akhir kali ini, penulis akan merancang sebuah *website* yang dapat digunakan untuk memprediksi kondisi dari Sungai Citarum. Dalam proses perancangan *website* ini penulis akan melakukan *deployment* pada model *machine learning* yang telah dibentuk menggunakan model layanan *cloud* yaitu *Platform-as-a-Service*. Layanan *Platform-as-a-Service* yang akan digunakan yaitu Heroku dan Microsoft Azure. Hasil dari Tugas Akhir ini yaitu membentuk *website* yang dapat digunakan oleh masyarakat umum dan melakukan pengujian terhadap performansi dari kedua *cloud server* untuk mendapatkan hasil berdasarkan parameternya yaitu *throughput*, *latency*, *error*, *CPU usage*, dan *memory usage*.

II. KAJIAN TEORI

A. Machine Learning

Machine learning merupakan salah satu cabang dari kecerdasan buatan (*artificial intelligence*). Secara singkat, *machine learning* adalah pemrograman komputer yang menggunakan data historis yang digunakan sebagai model pembelajaran untuk memberikan kinerja terbaik saat mengekstrak informasi dari kumpulan data. Model yang dibuat oleh *machine learning* didasarkan pada teori statistik yang bertujuan untuk menarik kesimpulan dari sampel data yang telah disediakan. Oleh karena itu, ada dua elemen penting yang ada dalam ilmu komputer. Pertama, *training* merupakan algoritma yang dibutuhkan untuk menyelesaikan sebuah masalah. Kedua, *testing* merupakan pengujian yang dilakukan dengan menggunakan nilai akurasi prediksi karena hasil yang mewakili model harus efisien dalam mencari sebuah solusi [8].

1. Random Forest Classification

Algoritma *random forest* merupakan salah satu algoritma yang

dapat menentukan prioritas atribut independent yang mempengaruhi atribut dependen. Algoritma ini telah diterapkan di banyak bidang, salah satunya adalah data mining. Secara umum, teknik *machine learning* sering memiliki formasi presisi yang tidak akurat. Tetapi keacakan ditentukan oleh pohon keputusan, sehingga algoritma *random forest* lebih akurat. *Random forest*, khususnya telah terbukti memiliki tingkat akurasi yang tinggi dalam memprediksi data dari berbagai atribut. Algoritma *random forest* membangun banyak pohon keputusan untuk mengikuti proses dan mendapatkan sebagian besar prediksi dari pohon-pohon ini [9].

B. Cloud Computing

Cloud computing adalah sebuah konsep untuk memberikan akses aman ke sumber daya dalam skala besar menggunakan kumpulan sumber daya komputer yang dapat dikonfigurasi, seperti jaringan, *server*, program, aplikasi, dan layanan. Beberapa hal ini juga dapat disampaikan dengan cepat dan mudah menggunakan upaya manajemen yang minimal atau interaksi dari penyedia layanan. Ada lima elemen penting dalam *cloud computing*, yaitu akses jarak jauh, elastisitas cepat, layanan aman, layanan yang disesuaikan dengan kebutuhan, dan pengumpulan sumber daya. Tidak hanya itu, *cloud computing* juga memiliki model layanan lain seperti *Platform-as-a-Service*, *Software-as-a-Service*, *Infrastructure-as-a-Service*, dan *Database-as-a-Service* [10].

1. Platform-as-a-Service

Platform-as-a-Service merupakan salah satu layanan utama dalam komputasi *cloud*. *Platform-as-a-Service* adalah pengiriman *platform* komputasi dan tumpukan solusi sebagai layanan tanpa unduhan atau instalasi perangkat lunak untuk melakukan pengembangan. *Platform-as-a-Service* dapat digunakan melalui penggunaan virtualisasi dan mekanisme dari berbagai sumber daya. Salah satu contohnya dengan menggunakan teknik virtualisasi, memungkinkan untuk membuka beberapa *platform Operation System* yaitu Linux, Ubuntu, Windows, dan lain sebagainya [11]. Penggunaan *Platform-as-a-Service* juga sudah kompleks karena sudah terdapat *database*, *operation system*, *web server*, dan *framework* dari aplikasi itu sendiri. Hal ini yang dapat memudahkan konsumen

dalam menggunakan *Platform-as-a-Service*.

C. Layanan Cloud Server

1. Microsoft Azure

Microsoft Azure adalah *platform cloud* fleksibel yang memungkinkan pengembangan aplikasi yang cepat, *debugging*, iterasi, dan manajemen lainnya di seluruh jaringan pusat data Microsoft. Microsoft Azure juga dapat dikembangkan menggunakan alat, bahasa pemrograman, atau kerangka kerja yang ada. Dimungkinkan untuk mengintegrasikan aplikasi *cloud public* dengan teknologi dan lingkungan informasi yang ada. Pembuatan *virtual machine* dilakukan melalui konsol manajemen, dimana pengguna memilih opsi yang berbeda dari daftar yang tersedia [12].

2. Heroku

Heroku adalah sebuah *platform* yang digunakan untuk membangun sebuah aplikasi web. Hal ini juga memungkinkan sebuah penerapan aplikasi yang diimplementasikan di sebagian besar teknologi kontemporer. Aplikasi pengguna dapat berjalan di dalam sebuah wadah ringan yang disebut dengan *dynos*. Sebuah aplikasi tunggal yang dapat menggunakan satu atau lebih dari jenis yang berbeda. Sebagaimana dinyatakan, permintaan HTTP yang dikirim ke dalam aplikasi yang akan di *hosting* menggunakan Heroku dan diterima oleh penyeimbang beban, yang akan diteruskan ke dalam satu set router yang disebut dengan mesh perutean. Hal ini dapat membentuk lapisan HTTP yang bertanggung jawab dalam menerima *quest* ulang dan meneruskannya ke *properdynos*. Kriteria pertama memilih *dynois* yang benar adalah untuk memastikan bahwa itu memiliki aplikasi yang permintaan dimaksud. Namun, jika aplikasi yang dilayani oleh lebih dari satu *dyno*, router HTTP secara acak memilih salah satu yang harus melayani permintaan tersebut. Tentu saja, ini bukanlah sebuah solusi yang memungkinkan pengguna sumber daya yang paling efisien [13].

D. Flask

Flask adalah kerangka kerja web yang menggunakan pemrograman Python dan diklasifikasikan sebagai jenis kerangka kerja mikro. Flask bertindak sebagai *framework* aplikasi dan tampilan web. Pengembang dapat

menggunakan Flask dan bahasa Python untuk membuat web terstruktur dan mengelola perilaku web dengan lebih mudah. Flask termasuk dalam sifat kerangka mikro karena tidak memerlukan alat atau Pustaka khusus untuk digunakan. Fitur dan komponen yang paling umum seperti validasi formular, *database*, dan lain-lain. *Micro-framework* di sini berarti Flask ingin membuat inti dari aplikasi sesederhana dan semudah mungkin untuk dikembangkan. Seperti yang diketahui fleksibilitas dan skalabilitas Flask sangat tinggi dibandingkan dengan *framework* lainnya [14].

E. Apache Jmeter

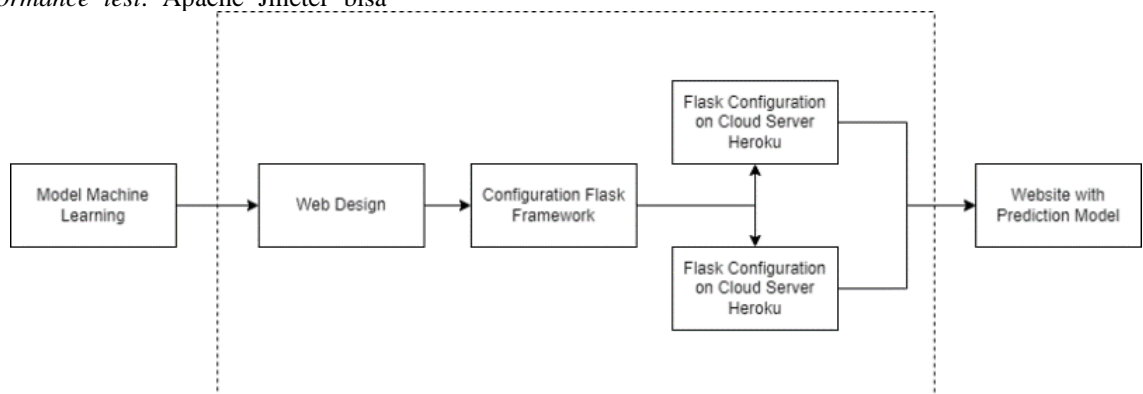
Aplikasi Apache Jmeter merupakan salah satu perangkat lunak *open-source* yang dijalankan dengan menggunakan Java murni dan dirancang untuk memuat suatu tes perilaku fungsional dan mengukur kinerja dari suatu *server*. Aplikasi ini dapat dilakukan untuk melakukan *performance test*. Apache Jmeter bisa

digunakan untuk melakukan *stress testing web application*, *FTP application*, dan *Database server test*. Jmeter dapat digunakan untuk menguji kinerja baik pada sumber daya statis dan dinamis. Hal ini dapat digunakan untuk mensimulasikan beban berat pada suatu *server*, sekelompok *server*, jaringan atau objek untuk menguji suatu kekuatan atau untuk menganalisis kinerja secara keseluruhan di bawah jenis beban yang berbeda [15].

III. METODE

A. Diagram Blok Penelitian

Dalam memahami penelitian secara terstruktur, maka akan diperlukan sebuah perancangan sistem yang baik dan benar. Perancangan sistem yang akan dilakukan adalah merancang website dan merancang model *machine learning*. Pada Gambar 1 menjelaskan sistem dalam bentuk diagram blok.

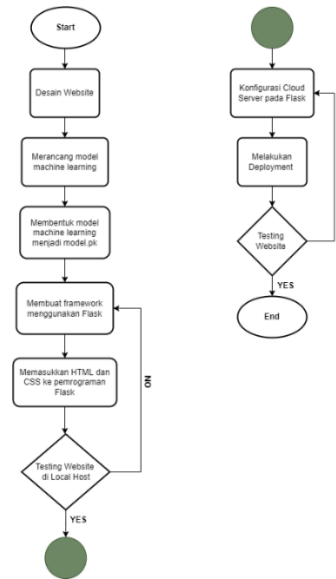


GAMBAR 1
DIAGRAM BLOK SISTEM YANG DIKEMBANGKAN

B. Flowchart Perancangan Website

Perancangan website kali ini akan dimulai dengan menggunakan *web framework* Flask. Bahasa pemrograman yang digunakan yaitu Python. Lalu, melakukan konfigurasi pada model *machine learning* yang sudah dibentuk dalam format *pickle* ke dalam *web framework* Flask. Setelah itu lakukan *testing* pada *local host* untuk mengetahui apakah konfigurasi Flask

yang dilakukan telah berhasil terbentuk dengan baik. Jika berhasil, maka aplikasi sudah siap untuk melanjutkan ke dalam tahap *deployment* menggunakan *cloud server*. Hasil akhir dari perancangan ini berupa sebuah *website* yang memiliki data dari *prediction machine learning model*. Pada Gambar 2 merupakan *flowchart* yang dapat menjelaskan perancangan *website* lebih terstruktur.



GAMBAR 2
FLOWCHART WEBSITE

C. Perancangan Model *Machine Learning*

Pada perancangan *machine learning* algoritma yang akan digunakan adalah *Random Forest Classification*. *Random Forest Classification* merupakan salah satu algoritma yang menciptakan sebuah pohon keputusan pada sampel yang berbeda dan

mengambil keputusan berdasarkan data mayoritas untuk memberikan hasil prediksi yang baik dengan tingkat akurasi yang tinggi. Tabel 1 memperlihatkan beberapa kategori yang akan dihasilkan dari proses pengolahan data menggunakan algoritma *Random Forest Clasification*.

TABEL 1
KATEGORI BANJIR

Kategori Label	Tinggi Air (meter)	Curah Hujan (milimeter)
Aman	≤ 5	≤ 50
Siaga 1	$5 < x \leq 6$	≤ 50
Siaga 1	≤ 5	$50 < x \leq 100$
Siaga 1	$5 < x \leq 6$	$50 < x \leq 100$
Siaga 2	> 6	$50 < x \leq 100$
Siaga 2	$5 < x \leq 6$	> 100
Siaga 2	> 6	> 100

D. Perangkat

1. Perangkat Keras

Tabel 2 merupakan spesifikasi Perangkat keras yang akan digunakan selama penelitian ini demi mendukung kinerja sistem adalah Laptop Asus VivoBook S14 sebagai berikut:

TABEL 2
SPESIFIKASI PERANGKAT KERAS

Spesifikasi Teknis	Nilai Spesifikasi Teknis
Sistem Operasi	Window 10 Home Single Language
Processor	Intel® Core™ i5-8250U CPU @ 1.60GHz 1.80 GHz
Penyimpanan	SSD Hitachi 128 GB, HDD Hitachi GHST 1 TB
Memori	8 GB

2. Perangkat Lunak

Perangkat lunak yang digunakan selama penelitian demi mendukung kinerja sistem serta spesifikasi adalah sebagai berikut:

- a. Pemrograman Python 3.9.1 untuk pembuatan *web framework* menggunakan Flask.
- b. Aplikasi Visual Studio Code sebagai *platform* untuk melakukan pemrograman.
- c. Apache Jmeter sebagai *platform* untuk melakukan perhitungan parameter pengujian.

Tabel 3 memperlihatkan spesifikasi dari *platform cloud server* yang digunakan demi mendukung kinerja sistem pada penelitian kali ini:

TABEL 3
SPESIFIKASI CLOUD SERVER

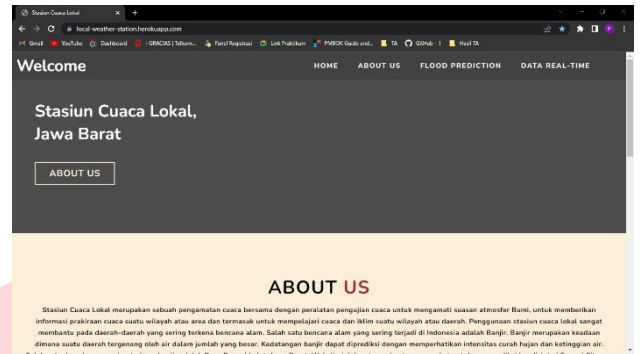
Layanan Cloud Server	Heroku	Microsoft Azure
RAM	512 MB	1 GB
Penyimpanan	Menggunakan <i>add-on</i>	5 GB
vCPU	1 vCPU	1 vCPU
Lokasi Penempatan	Global	Global
Free Tier	Selalu	12 Bulan

IV. HASIL DAN

PEMBAHASAN

A. Hasil Website

Tugas Akhir kali ini menggunakan model *machine learning* dengan algoritma *random forest*. Gambar 3 merupakan model yang telah sampai di tahap *deployment* ke dalam sebuah website akan terbentuk sebagai berikut:

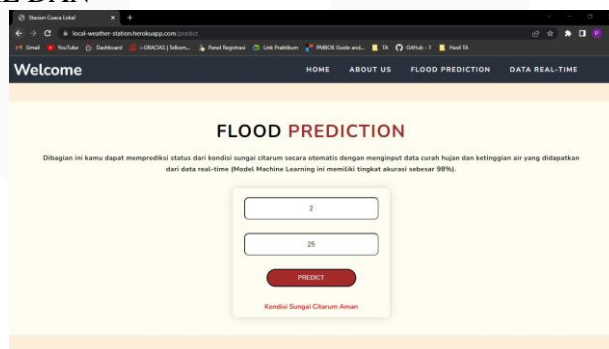


GAMBAR 3
DASHBOARD WEBSITE

Ada pun hasil pengujian pada fitur prediksi yang telah dibentuk yaitu sebagai berikut:

1. Aman

Pada Gambar 4 eksperimen akan dilakukan dengan pengujian kondisi aman, yang mana penulis akan mencoba memasukkan data ketinggian dengan nilai 2 meter dan intensitas curah hujan 25 milimeter. Maka hasil dari prediksi adalah aman.

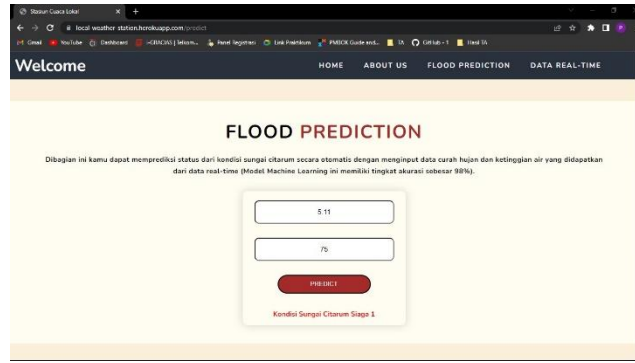


GAMBAR 4
KONDISI AMAN

2. Siaga 1

Pada Gambar 5 eksperimen akan dilakukan dengan pengujian kondisi siaga 1, yang mana penulis akan mencoba memasukkan data

ketinggian dengan nilai 5.11 meter dan intensitas curah hujan 75 milimeter. Maka hasil dari prediksi adalah Siaga 1.

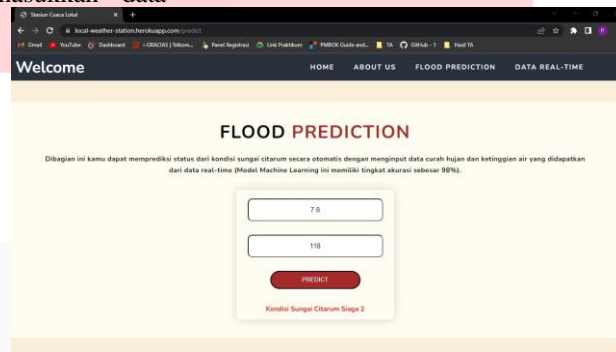


GAMBAR 5
KONDISI SIAGA 1

3. Siaga 2

Pada Gambar 6 eksperimen akan dilakukan dengan pengujian kondisi siaga 2, yang mana penulis akan mencoba memasukkan data

ketinggian dengan nilai 7.8 meter dan intensitas curah hujan 118 milimeter. Maka hasil dari prediksi adalah Siaga 2.



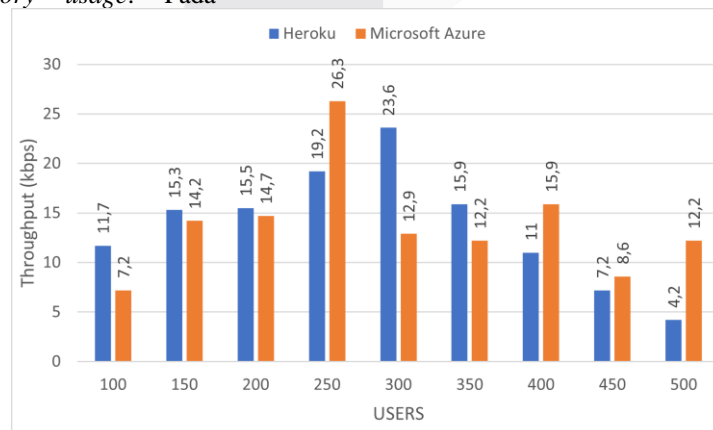
GAMBAR 6
KONDISI SIAGA 2

B. Hasil Pengujian Cloud Server

Hasil pengujian cloud server kali ini dilakukan untuk membandingkan kedua cloud server berdasarkan parameter yang akan digunakan yaitu throughput, latency, load time, CPU usage, dan memory usage. Pada

pengujian kali ini akan diberikan beban user sebanyak 100, 150, 200, 250, 300, 350, 400, 450, dan 500 user, ramp-up periode sebesar 5 second, dan loop count sebesar 1 kali.

1. Throughput



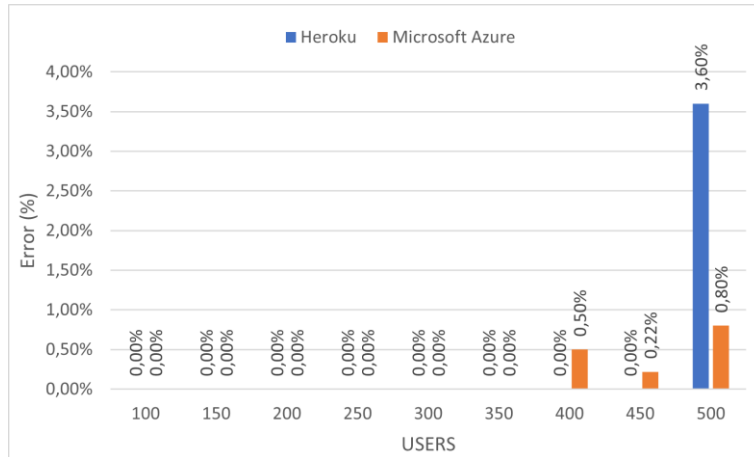
GAMBAR 7
GRAFIK THROUGHPUT CLOUD SERVER

Pada Gambar 7 di atas hasil pengujian ke-5 pada cloud server

Heroku dengan jumlah user 300 mendapatkan hasil throughput sebesar

23.6 kbps dan pada *cloud server* Microsoft Azure pengujian ke-4 dengan jumlah *user* 250 mendapatkan hasil *throughput* sebesar 26.3 kbps.

2. Error

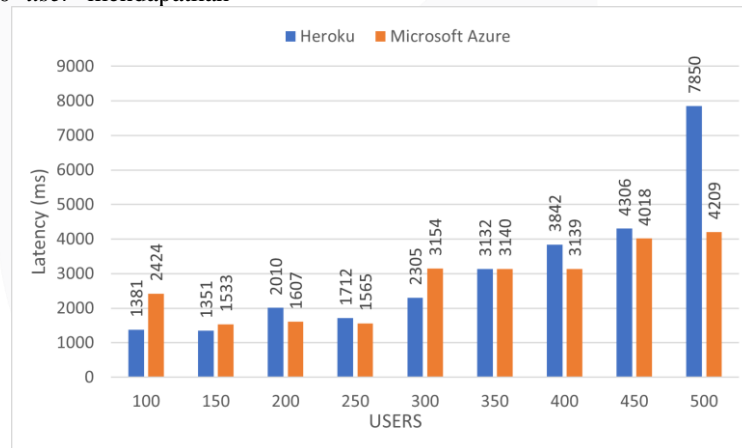


GAMBAR 8
GRAFIK ERROR CLOUD SERVER

Pada Gambar 8 di atas hasil pengujian ke-9 pada *cloud server* Heroku dengan jumlah *user* 500 mendapatkan hasil *error* sebesar 3.60% dan pada *cloud server* Microsoft Azure pengujian 7, pengujian 8, dan pengujian 9 dengan jumlah *user* 400 *user*, 450 *user*, dan 500 *user* mendapatkan

hasil *error* sebesar 0.50%, 0.22%, dan 0.80%.

3. Latency

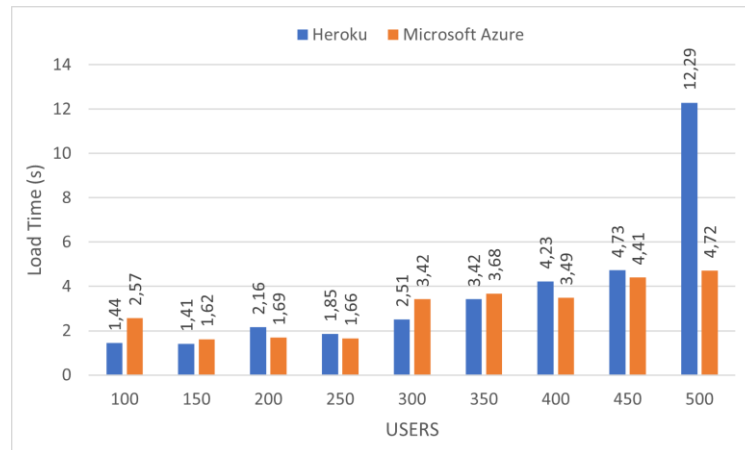


GAMBAR 9
GRAFIK LATENCY CLOUD SERVER

Pada Gambar 9 di atas hasil pengujian berdasarkan dengan batasan masalah mendapatkan *latency* kecil dari 2500 ms maka ditarik kesimpulan bahwa pada *cloud server* Heroku besar *latency* yang baik terdapat pada pengujian ke-1, pengujian ke-2, pengujian ke-3, pengujian ke-4, dan

pengujian ke-5. Pada *cloud server* Microsoft Azure besar *latency* yang baik terdapat pada pengujian ke-1, pengujian ke-2, pengujian ke-3, dan pengujian ke-4.

4. Load Time

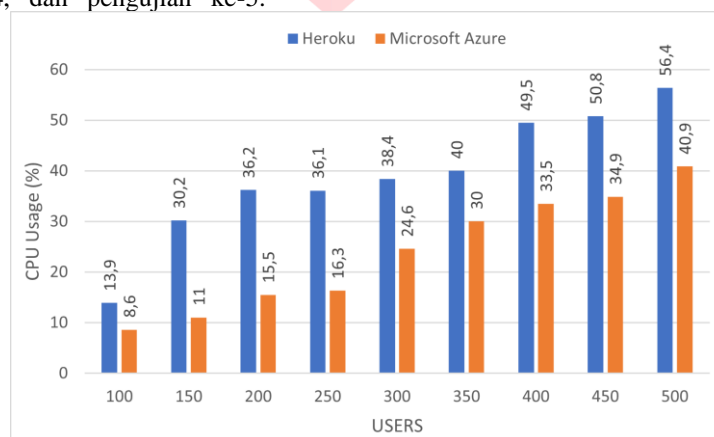


GAMBAR 10
GRAFIK LOAD TIME CLOUD SERVER

Gambar 10 di atas menunjukkan hasil pengujian berdasarkan dengan batasan masalah *load time* yang baik kecil dari 3 second, sehingga ditarik kesimpulan bahwa pada *cloud server* Heroku besar *load time* yang baik terdapat pada pengujian ke-1, pengujian ke-2, pengujian ke-3, pengujian ke-4, dan pengujian ke-5.

Pada *cloud server* Microsoft Azure besar *load time* yang baik terdapat pada pengujian ke-1, pengujian ke-2, pengujian ke-3, dan pengujian ke-4.

5. CPU Usage

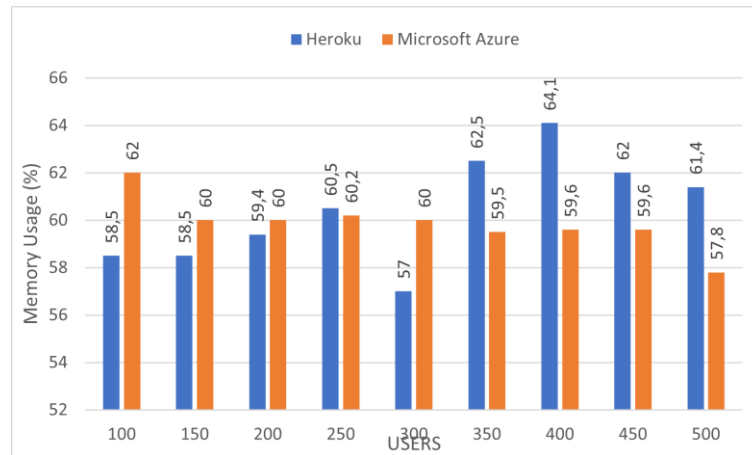


GAMBAR 11
GRAFIK CPU USAGE

Semakin banyak *user* maka akan semakin besar jumlah *CPU usage*. Gambar 11 di atas menunjukkan hasil pengujian, yang mana pada *cloud server* Heroku penggunaan *CPU usage* paling tinggi berada pada pengujian ke-9 dengan jumlah *user* 500 yaitu sebesar

56.4% dan pada *cloud server* Microsoft Azure penggunaan *CPU usage* pa ling tinggi berada pada pengujian ke-9 dengan jumlah *user* 500 yaitu sebesar 40.9%.

6. Memory Usage



GAMBAR 12
GRAFIK MEMORY USAGE CLOUD SERVER

Gambar 12 menunjukkan bahwa *memory usage* berada dalam kondisi stabil. Pada *cloud server* Heroku penggunaan *memory usage* terlihat stabil dengan hasil pengujian berada di antara 59.6% hingga 60% dan pada *cloud server* Microsoft Azure penggunaan *memory usage* terlihat stabil dengan hasil pengujian berada diantara 59.5% hingga 60.2%.

V. KESIMPULAN

Implementasi model *machine learning* menggunakan layanan *cloud* yaitu *Platform-as-a-Service* dengan menggunakan dua *cloud server* yaitu Microsoft Azure dan Heroku berhasil dilakukan. Model *machine learning* yang dibentuk menggunakan algoritma *random forest classification* dapat memprediksi suatu kondisi dari Sungai Citarum yang menjadi tiga kategori yaitu aman, siaga 1, dan siaga 2. Model yang sudah dibentuk dapat langsung diinstal pada dua *platform cloud server* yaitu Heroku dan Microsoft Azure. Parameter yang digunakan sebagai pembandingan antara kedua *cloud server* tersebut yaitu *throughput*, *latency*, *load time*, *CPU usage*, dan *memory usage*. Hasil eksperimen kedua *cloud server* tersebut menunjukkan bahwa Heroku merupakan *cloud server* yang lebih layak diimplementasikan dalam melakukan *deployment model machine learning* dibandingkan dengan Microsoft Azure. Pengujian *cloud server* yang sangat baik terdapat pada pengujian ke-5 dengan jumlah user 300 pada *cloud server* Heroku. Pengujian ke-5 Heroku mendapatkan *throughput* sebesar 23.6 kbps dengan *error* 0.00%, *latency* sebesar 2305 ms, dan *load time* sebesar 2.5 second. Sedangkan, untuk *cloud server* Microsoft Azure yang sangat baik terdapat pada pengujian ke-

4 dengan jumlah user 250 pada *cloud server* Microsoft Azure. Pengujian ke-4 Microsoft Azure mendapat *throughput* sebesar 26.3 kbps dengan *error* 0.00%, *latency* sebesar 1565 ms, dan *load time* sebesar 1.66 second.

REFERENSI

- [1] L. Baier, S. Seebacher, And R. Paper Baier, "Challenges In The Deployment And Operation Of Machine Learning In Practice." [Online]. Available: <https://www.researchgate.net/publication/332996647>
- [2] J. Carreira, P. Fonseca, A. Tumanov, A. Zhang, And R. Katz, "Cirrus: A Serverless Framework For End-To-End ML Workflows," *Soc 2019 - Proc. Acm Symp. Cloud Comput.*, Pp. 13–24, 2019, Doi: 10.1145/3357223.3362711.
- [3] S. Tuli, S. Tuli, R. Tuli, And S. S. Gill, "Predicting The Growth And Trend Of Covid-19 Pandemic Using Machine Learning And Cloud Computing," *Internet Of Things (Netherlands)*, Vol. 11, 2020, Doi: 10.1016/J.Iot.2020.100222.
- [4] W. Hassan, T.-S. Chou, O. Tamer, J. Pickard, P. Appiah-Kubi, And L. Pagliari, "Cloud Computing Survey On Services, Enhancements And Challenges In The Era Of Machine Learning And Data Science," *Int. J. Informatics Commun. Technol.*, Vol. 9, No. 2, P. 117, Aug. 2020, Doi: 10.11591/Ijict.V9i2.Pp117-139.
- [5] Q. Xia, Y. Lan, And L. Xiao, "The Status Prediction Of Physical Machine In IaaS Cloud Environment," In *Proceedings -*

- 2015 *International Conference On Cyber-Enabled Distributed Computing And Knowledge Discovery, Cyberc 2015*, Oct. 2015, Pp. 302–305. Doi: 10.1109/Cyberc.2015.100.
- [6] W. Sun, K. Zhang, S. K. Chen, X. Zhang, And H. Liang, “Software As A Service: An Integration Perspective,” *Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, Vol. 4749 Lncs, Pp. 558–569, 2007, Doi: 10.1007/978-3-540-74974-5_52.
- [7] M. Boniface *Et Al.*, “Platform-As-A-Service Architecture For Real-Time Quality Of Service Management In Clouds,” *5th Int. Conf. Internet Web Appl. Serv. Iciw 2010*, Pp. 155–160, 2010, Doi: 10.1109/Iciw.2010.91.
- [8] “Implementasi Machine Learning Prediksi Harga Sewa Apartemen Menggunakan Algoritma Random Forest Melalui Framework Website Flask Python.”
- [9] P. Prihandoko, B. Bertalya, And L. Setyowati, “City Health Prediction Model Using Random Forest Classification Method,” *2020 5th Int. Conf. Informatics Comput. Icic 2020*, 2020, Doi: 10.1109/Icic50835.2020.9288542.
- [10] T. Dillon, C. Wu, And E. Chang, “Cloud Computing: Issues And Challenges,” *Proc. - Int. Conf. Adv. Inf. Netw. Appl. Aina*, Pp. 27–33, 2010, Doi: 10.1109/Aina.2010.187.
- [11] W. Tian, S. Su, And G. Lu, “A Framework For Implementing And Managing Platform As A Service In A Virtual Cloud Computing Lab,” *2nd Int. Work. Educ. Technol. Comput. Sci. Etc. 2010*, Vol. 2, No. 60904072, Pp. 273–276, 2010, Doi: 10.1109/Etc.2010.126.
- [12] B. S. Dordević, S. P. Jovanović, And V. V. Timčenko, “Cloud Computing In Amazon And Microsoft Azure Platforms: Performance And Service Comparison,” *2014 22nd Telecommun. Forum, Telfor 2014 - Proc. Pap.*, Pp. 931–934, 2014, Doi: 10.1109/Telfor.2014.7034558.
- [13] P. Danielsson, T. Postema, And H. Munir, “Heroku-Based Innovative Platform For Web-Based Deployment In Product Development At Axis,” *Ieee Access*, Vol. 9, Pp. 10805–10819, 2021, Doi: 10.1109/Access.2021.3050255.
- [14] P. Adi Nugroho, I. Fenriana, And R. Arijanto, “Implementasi Deep Learning Menggunakan Convolutional Neural Network (Cnn) Pada Ekspresi Manusia,” *J. Algor*, Vol. 2, No. 1, 2020, [Online]. Available: <https://jurnal.buddhidharma.ac.id/Index.Php/Algor/Index>
- [15] F. Fandy, R. Rosmasari, And G. M. Putra, “Pengujian Kinerja Web Server Atas Penyedia Layanan Elastic Cloud Compute (Ec2) Pada Amazon Web Services (Aws),” *Adopsi Teknol. Dan Sist. Inf.*, Vol. 1, No. 1, Pp. 21–35, 2022.