

Implementasi Widgets Builder untuk Monitoring Kinerja Sistem Komputer dengan Menggunakan Rainmeter

Alexander Gracetantiono¹, Budi Wasito²

Program Studi Teknik Informatika
Institut Bisnis dan Informatika Kwik Kian Gie
Jalan Yos Sudarso Kav.87 Sunter, Jakarta Utara 14350

¹alexandergracetantiono@gmail.com, ²budi.wasito @kwikkiangie.ac.id

ABSTRACT

The study goal is to create a widget builder application with the feature to generate and import INI files that can be run on the Rainmeter application. The purpose of implementing a widget builder is to assist computer users in making widgets that can be used to monitor system performance. This study refers to the opinion of Sulasno and Rakhmat, which is a monitoring system for the data collection process as well as analyzing the use of limited computer resources such as storage memory, central processing unit, random access memory, and various other computer resources. This study uses a personal extreme programming method that prioritizes delivery speed and a team consisting of one person, data is collected by questionnaire method, observation method and documentation method which has Rainmeter users as respondents. In the widget builder, there are several main features such as an image upload feature, an additional feature for measuring computer resource usage, and a save widget feature. This feature obtained from research conducted by the author is a solution to help Rainmeter users so that widget creation can be done without coding. From the results of the research, it is concluded that the features in the widget builder application that have been implemented have helped Rainmeter users.

Keywords: Rainmeter, Widget Builder, System Monitoring, Personal Extreme Programming.

1. PENDAHULUAN

Dengan perkembangan yang pesat ini, pemakaian sumber daya seperti *random access memory* (RAM), dan *central processing unit* (CPU) semakin dibutuhkan dalam jumlah yang cukup besar walau hanya untuk menjalankan perangkat lunak aplikasi. Pemakaian sumber daya komputer ini perlu dipantau agar pengguna mengetahui batas maksimum pemakaian sumber daya komputer. Selain itu pemantauan penggunaan jaringan internet juga diperlukan untuk mengetahui aktivitas *upload* dan *download* pada computer.

Dalam berbagai kasus, ketika membuka banyak aplikasi dalam satu waktu, terkadang menyebabkan beberapa aplikasi lainnya menjadi *not responding*, hal ini biasanya disebabkan oleh

penggunaan sumber daya komputer yang sudah maksimal sehingga menyebabkan perlambatan pada sistem komputer.

Oleh karena itu aplikasi pemantauan pemakaian sumber daya sistem yang tidak menyebabkan sistem menjadi *not responding*, mudah serta cepat untuk diakses sangat diperlukan, terutama untuk pengguna komputer dengan spesifikasi *hardware* komputer atau laptop yang kurang memadai.

2. METODE PENELITIAN

Dalam pembuatan penelitian yang akan dilakukan, penulis memilih untuk menggunakan metode SDLC *personal extreme programming* karena metode tersebut memiliki keunggulan yaitu memiliki tim yang relatif sedikit serta

pengembangan aplikasi dalam jangka waktu sebentar sehingga dapat mempercepat proses penulisan skripsi penulis dan pengembangan aplikasi.

Proses dalam SDLC tersebut yang digunakan penulis meliputi tahap – tahapan berikut, yaitu tahap *planning*, tahap desain, tahap pengerjaan dan tahap pengujian. Tahap *planning* meliputi pembuatan latar belakang, penentuan identifikasi masalah, penentuan batasan masalah, pencarian teori dasar dan sistem berjalan yang berhubungan, dari tahap *planning* ini penulis dapat menemukan fitur yang dibutuhkan untuk melakukan pemantauan penggunaan sumber daya komputer dan kesulitan yang dialami oleh pengguna Rainmeter.

Tahap berikutnya, yaitu tahap desain yang meliputi perancangan arsitektur sistem, pembuatan diagram alur yang baru, pembuatan *use case diagram*, pembuatan diagram aktivitas, dan perancangan tampilan struktur menu. Pada tahap pengembangan aplikasi, penulis melakukan proses pengkodean sesuai dengan desain yang telah dibuat. Pada tahap terakhir, penulis melakukan pengujian aplikasi, dimana aplikasi akan dirilis di forum Rainmeter untuk memastikan bahwa aplikasi dapat memecahkan masalah yang dialami oleh pengguna Rainmeter.

Metode pengembangan *personal extreme programming* merupakan metode pengembangan yang dinamis akan perubahan, dan dapat dipakai pada pengembangan aplikasi dengan jumlah orang yang sedikit. tahap – tahap yang dilalui adalah:

- Planning/Perencanaan. Pada tahap ini, dilakukan proses memahami proses dan menentukan input serta output dari aplikasi serta fitur-fitur pada aplikasi.
- Design/Perancangan. Pada tahap ini dilakukan perancangan system.
- Coding/Pengkodean. Pada tahap ini dilakukan pembuatan aplikasi.
- Testing/Pengujian. Pada tahap ini dilakukan pengujian terhadap aplikasi yang dibuat.



Gambar 1. Metode Pengembangan Extreme Programming

3. LANDASAN TEORI

System Monitoring

Pengertian *monitoring* atau pemantauan menurut Fietri dan Ilham (2021:25) Monitoring adalah sebuah siklus kegiatan yang meliputi proses pengumpulan, peninjauan ulang pelaporan dan tindakan atas informasi suatu proses yang sedang diimplementasikan. Sedangkan pengertian *system monitoring* menurut Sulasno dan Rakhmat (2020:2), pemantauan adalah proses pengumpulan data serta melakukan analisis terhadap pemakaian sumber daya komputer terbatas seperti memori penyimpanan, *central processing unit*, *random access memory*, *graphic card* *Virtual RAM*, dan berbagai sumber daya komputer lainnya. Proses pemantauan diperlukan untuk menganalisa apakah sumber daya komputer masih cukup layak untuk digunakan atau memerlukan penambahan kapasitas.

Tujuan pemantauan sumber daya komputer adalah untuk mengumpulkan informasi yang berguna dari berbagai bagian komputer sehingga sumber daya computer dapat diatur dan dikontrol dengan menggunakan informasi yang telah terkumpul. Dengan begitu diharapkan apabila terjadi suatu masalah dalam *system computer* akan cepat diketahui dan diperbaiki sehingga stabilitas *system computer* lebih terjamin.

Sumber Daya Sistem

Pengertian sumber daya sistem Sulasno dan Rakhmat (2020:1), sumber daya sistem komputer adalah komponen fisik atau *virtual* yang memiliki ketersediaan yang terbatas dalam sistem komputer. Ketersediaan sumber daya

komputer sangat diperlukan untuk menjaga kinerja sistem komputer supaya dapat melakukan pengoperasian perangkat lunak secara maksimal.

Sumber daya komputer menurut Pragma dan Manjusha (2016: 4095) sumber daya komputer terbagi menjadi

1. *Compute resources* atau sumber daya komputasi biasanya merupakan kumpulan mesin fisik, masing-masing terdiri dari satu atau lebih prosesor memori, antarmuka jaringan, dan I/O lokal, yang semuanya bersama-sama menyediakan kapasitas komputasi atau perhitungan untuk pengguna komputer.

2. *Networking Resources* atau sumber daya jaringan, memiliki fungsi untuk menghitung sumber daya pada mesin fisik dalam pusat data yang dikemas ke dalam rak dan biasanya diatur sebagai kelompok dari ribuan host untuk tujuan alokasi sumber daya. Topologi jaringan pusat data saat ini didasarkan pada hierarki, topologi

seperti pohon yang serupa dengan yang digunakan pada jaringan telepon awal, meskipun sejumlah topologi alternatif termasuk proposal berdasarkan pohon gemuk, hyper-cube, dan topologi dunia kecil acak telah muncul.

3. *Storage resources* atau sumber daya penyimpanan merupakan suatu alat penyimpanan persisten yang memiliki beberapa jenis, mulai dari disk virtual dan layanan basis data hingga penyimpanan objek, setiap layanan memiliki tingkat jaminan dan keandalan konsistensi data yang berbeda.

4. *Power resources* atau sumber daya listrik merupakan banyaknya daya dikonsumsi oleh komputer, daya juga diperlukan untuk peralatan jaringan, peralatan distribusi daya, peralatan *input* dan *output*, serta pendinginan dan infrastruktur pendukung.

Widgets

Menurut D'Aloisio (2019:41) menjelaskan *Widgets* adalah aplikasi perangkat lunak yang menampilkan informasi dan melakukan berbagai jenis fungsi pada *desktop* komputer di sebuah perangkat komputasi. *Widget* juga dapat disebut sebagai *dashboard*, papan *widget*, ataupun *gadget*. Pada sistem operasi windows, misalnya, *desktop widget* memiliki panggilan lain yaitu disebut sebagai *gadget desktop*, *gadget desktop* ini dapat digunakan untuk menampilkan waktu, cuaca, kalender, atau informasi lain pada layar *desktop* perangkat komputasi.

Pada perangkat sistem operasi android, *widget desktop* memiliki fungsi yang hampir

sama dengan *widget* pada sistem operasi windows, yaitu digunakan untuk menampilkan informasi seperti waktu, catatan, notifikasi dan cuaca, *widget* ini ditampilkan pada layar beranda perangkat komputasi mobile, seperti tablet komputer, telepon pintar, atau perangkat seluler lainnya.

Rainmeter

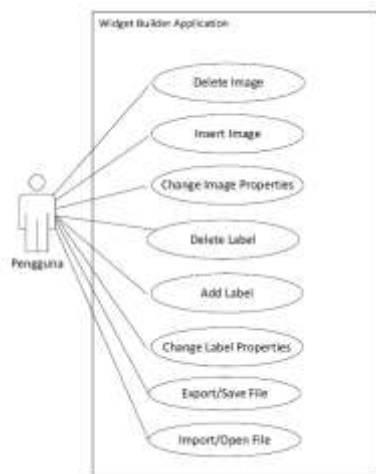
Pengertian Rainmeter menurut website resmi Rainmeter, adalah utilitas kustomisasi desktop *open-source* dan gratis untuk Windows, dirilis di bawah lisensi GNU GPL v2. Hal ini memungkinkan pengguna untuk membuat dan menampilkan *widget* atau *applet desktop* yang disebut "*skins*" yang dapat dibuat dan disesuaikan oleh pengguna. Koleksi skin yang siap digunakan dapat diunduh dan diinstal dalam paket yang dikenal sebagai "*suites*".

4. HASIL DAN PEMBAHASAN

a. Desain Sistem

Sistem yang dibangun adalah sebuah sistem pembuatan *widget* pada sistem operasi windows. Dimana sistem ini digunakan untuk memudahkan pengguna Rainmeter untuk berkreasi dalam pembuatan sebuah *widget* untuk melakukan pemantauan penggunaan sumber daya komputer secara *real-time*. Pengguna aplikasi juga tidak perlu memiliki keahlian dalam melakukan pengkodean, selain itu aplikasi juga dilengkapi *user guide* yang diharapkan dapat membantu pengguna baru dalam memakai aplikasi *widget builder* ini. Aplikasi *widget builder* ini memiliki fungsi untuk memasukkan gambar, teks, dan pengukuran penggunaan *central processing unit* (CPU), *random access memory* (RAM), *upload* dan *download speed*, serta melakukan penyimpanan *widget* dan *import widget* dari file yang sudah pernah disimpan. File *widget* yang dihasilkan dapat di jalankan menggunakan aplikasi Rainmeter, tujuannya agar *widget* hasil buatan pengguna dapat ditampilkan pada layar *desktop* pengguna.

b. Use Case Diagram



Gambar 2. Use Case Diagram Widget Builder Application

Use Case Diagram Widget Builder Application, terdapat satu aktor yaitu pengguna yang menggunakan aplikasi *widget builder*. Pengguna dapat melakukan *insert image*, menambahkan pengukuran penggunaan sumber daya komputer, mengubah property dari gambar dan teks, menyimpan hasil pengeditan, serta melakukan *import* dari project yang pernah disimpan.

- (1) *Use case insert image*. adalah *use case* untuk melakukan *insert image* pada aplikasi yang dibuat oleh penulis. Pengguna dapat memasukkan gambar sebagai sebagai latar atau tampilan yang akan ditampilkan pada layar desktop dengan bantuan aplikasi Rainmeter.
- (2) *Use case insert label* adalah *use case* untuk melakukan *insert label* pada aplikasi yang dibuat oleh penulis. Pengguna dapat menambahkan label yang berfungsi untuk menampilkan informasi seperti penggunaan RAM, CPU, dan jaringan pada komputer, selain itu pengguna juga dapat memilih label kustom untuk diberikan teks atau tulisan bebas.
- (3) *Use case change image properties* adalah *use case* untuk melakukan perubahan properti dari sebuah gambar, pengguna dapat melakukan perubahan properti seperti koordinat dari gambar, lebar, dan tinggi gambar.
- (4) *Use case change label properties* adalah *use case* untuk melakukan perubahan properti dari sebuah label, pengguna dapat melakukan perubahan properti seperti koordinat dari

label, lebar, tinggi, warna teks, dan ukuran font.

- (5) *Use case delete image*, pada *use case* ini pengguna *widget builder* dapat melakukan penghapusan gambar dari *canvas*, gambar yang dihapus tidak akan mempengaruhi file asli gambar.
- (6) *Use case delete label*, pada *use case* ini pengguna *widget builder* dapat melakukan penghapusan label dari *canvas*.
- (7) *Use case save file*, pada *use case* ini pengguna *widget builder* dapat melakukan penyimpanan file. File yang disimpan akan berbentuk file konfigurasi yang dapat dijalankan pada Rainmeter.
- (8) *Use case open file*, pada *use case* ini pengguna *widget builder* dapat membuka file dari hasil pembuatan atau pengeditan yang sudah disimpan. File yang dibuka akan dapat diedit dan disimpan pada tempat yang berbeda.

c. Aplikasi

Aplikasi *widget builder* yang dibuat diberi nama “Creather” yang merupakan penggabungan kata dari *Create* dan *Weather*, *weather* atau cuaca sendiri dipilih karena Rainmeter yang memiliki kata *rain* atau hujan merupakan bagian dari cuaca. Berikut ini adalah hasil program yang telah dibuat berdasarkan rancangan program.

- (1) Halaman *Laucher* (Menu Utama)



Gambar 3. Halaman *Laucher*

Gambar 3 halaman *launcher* yang ditampilkan pertama kali ketika aplikasi dijalankan oleh pengguna. Pada halaman ini pengguna dapat memilih untuk memulai aplikasi (*Launch Application*), *About*, dan *Help*.

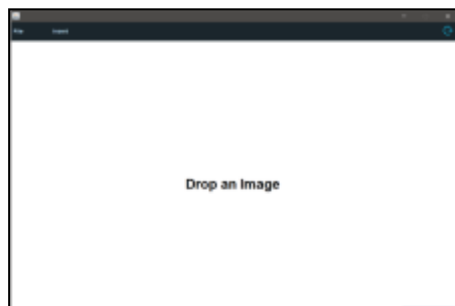
(2) Halaman *user guide*



Gambar 4. Halaman *user guide*

Gambar 4 merupakan halaman launcher ketika pilihan *Help* ditekan, akan ditampilkan bantuan berupa *step by step* penggunaan aplikasi Creather.

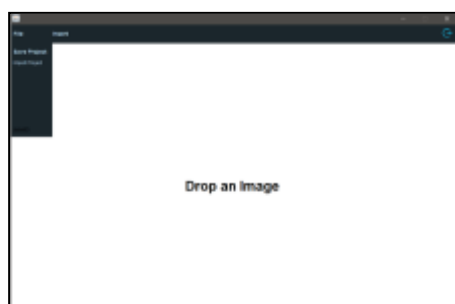
(3) Halaman Canvas



Gambar 5. Halaman Canvas

Gambar 5 merupakan tampilan halaman *canvas*, pada halaman ini pengguna dapat melakukan *drop* gambar, dan melakukan pengeditan *label* dan gambar.

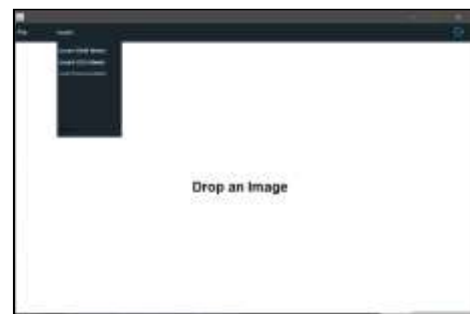
(4) Pemilihan Toolbar File



Gambar 6. Tampilan Pilihan Toolbar File

Gambar 6 menampilkan pilihan saat *toolbar file* ditekan, pilihan meliputi *save project* dan *import project*.

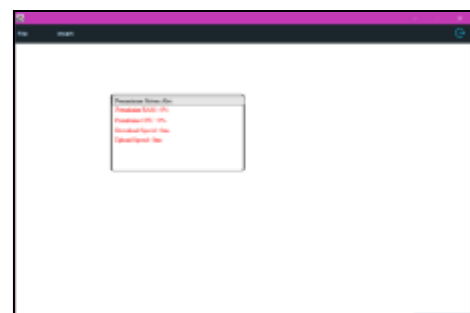
(5) Pemilihan Toolbar Insert



Gambar 7. Tampilan Pilihan Insert file

Gambar 7 menampilkan pilihan saat *toolbar Insert* ditekan, pilihan meliputi *insert RAM meter*, *insert CPU meter*, dan *insert download meter*.

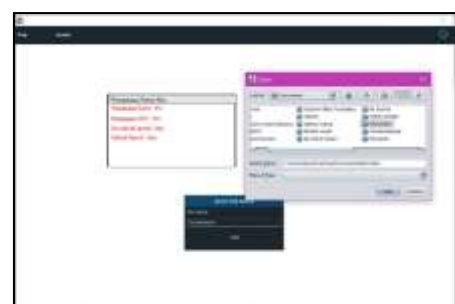
(6) Tahap Pembuatan Widget



Gambar 8. Tampilan Pembuatan Widget

Gambar 8 menampilkan ketika gambar di *drop* pada *canvas*, pada gambar terdapat dua gambar dan dua *label*, *label* dapat diambil dari *Insert -> Label*.

(7) Tahapan Penyimpanan File



Gambar 9. Tampilan Penyimpanan File

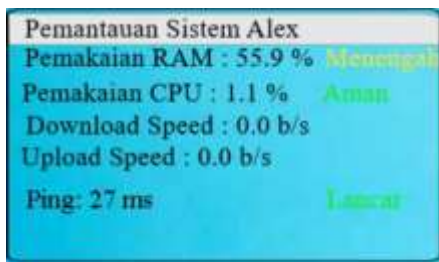
Gambar 9 akan ditampilkan folder tempat penyimpanan yang dapat dipilih. Penulis menyimpan pada alamat "C:\Users\alexa\Documents\Rainmeter"

(8) Hasil Pemantauan Desktop



Gambar 10. Hasil pemantauan penggunaan sumber daya komputer pada desktop

(9) Hasil Pemantauan Desktop Detail



Gambar 11. Hasil pemantauan penggunaan sumber daya komputer detail

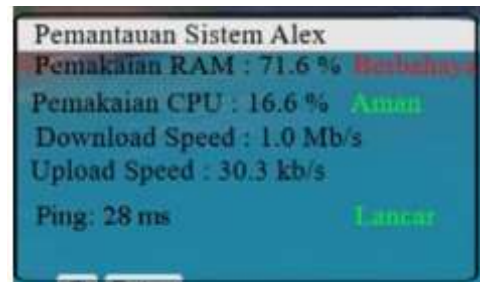
Gambar 10 dan gambar 11 merupakan hasil tampilan uji coba aplikasi, gambar, label dan posisi masing objek sama dengan yang ditampilkan pada *canvas*.

(10) Hasil Pemantauan Pada Gim



Gambar 11. Hasil pemantauan penggunaan sumber daya computer pada sebuah gim

(11) Hasil Detail Pemantauan Pada Gim



Gambar 12. Detail Hasil pemantauan penggunaan sumber daya computer pada sebuah gim

Gambar 11 dan gambar 12 merupakan hasil tampilan uji coba *monitoring* pemakaian CPU dan RAM pada sebuah gim windows.

PEMBAHASAN

Pada gambar 10 dapat terlihat bahwa penggunaan RAM laptop sebanyak 55.9 persen dari jumlah total RAM, pemakaian CPU sebanyak 1.1 persen, dan memiliki ping sebesar 27ms yang artinya lancar. Apabila penggunaan RAM melebihi 50 persen widget akan memberitahukan bahwa penggunaan RAM masih diambang batas normal, pemakaian cpu dibawah 50 persen akan dituliskan “aman”, dan ping dibawah 99ms masih dikategorikan lancar.

Pada gambar 11 sebuah aplikasi gim dijalankan untuk mengetahui jumlah pemakaian sumber daya sistem dipakai oleh sebuah gim, dapat dilihat bahwa pemakaian RAM pada laptop meningkat sebanyak 28 persen, maka pemakaian RAM menjadi 71.6 persen dari total kapasitas RAM laptop. Pemakaian RAM yang diatas 60 persen tidak disarankan oleh penulis karna akan menyebabkan panas berlebih pada laptop penulis, oleh karena itu penggunaan RAM diatas 60 persen akan dikategorikan sebagai berbahaya.

Pemakaian CPU juga meningkat menjadi 16.6 persen, pemakaian CPU yang dibawah 50 persen masih dapat dikategorikan aman, berdasarkan pengalaman penulis bermain gim tidak akan banyak mempengaruhi pemakaian CPU pada laptop penulis.

Koneksi jaringan juga dapat dipantau dengan widget yang sudah dibuat dengan menggunakan aplikasi penulis, dapat dilihat bahwa koneksi penulis cukup stabil, penulis melakukan ping ke www.google.com dan

mendapati 27 ms yang dapat dikatakan lancar

SIMPULAN

Berdasarkan hasil pengumpulan data melalui angket yang disebarakan pasca penelitian penulis dapat menyimpulkan beberapa hal yaitu.

Widget yang dibuat menggunakan aplikasi widget builder milik penulis, dapat melakukan pemantauan penggunaan sumber daya komputer seperti pemakaian RAM, CPU, dan jaringan internet. Widget juga dilengkapi dengan pemberitahuan tentang pemakaian sumber daya komputer agar pengguna dapat menindak permasalahan seperti dengan melakukan penambahan atau penggantian perangkat penyedia sumber daya komputer (RAM, CPU).

Aplikasi Widget Builder untuk Rainmeter yang penulis implementasikan dapat menerima file berupa gambar dan dapat ditambahkan teks untuk memantau pemakaian CPU, RAM, dan kecepatan download. Pengguna dapat melakukan perubahan posisi gambar maupun label yang nanti hasil kerja pada desktop akan sama persis dengan yang dikerjakan oleh pengguna.

Pembuatan widget yang sebelumnya memerlukan pengkodean, dengan aplikasi widget builder akan dapat membantu pengguna yang kesusahan dalam melakukan pengkodean.

DAFTAR PUSTAKA

- Aloisio, N. D., Liu, A., Huang, M., & Song, M. (2019). SYSTEMS AND METHODS FOR DELIVERING TASK - ORIENTED CONTENT USING A DESKTOP WIDGET. *US 10,503,357 B2*, 41.
- Bourne, K. C. (2017). *Application Administrators Handbook Installing, Updating, and Troubleshooting Software* (Vol. 5). Waltham: Elsevier.
- Cabrera, J. (2017). *Modular Design Frameworks*. New York, United States of America: Holbrook.
- Fukala, M. (2017). Application Builder with Automated Data Objects Creation. *US 2017/0115968 A1*, 1-18.
- GNU Corporation. (2019, Februari 21). *Categories of free and nonfree software*. Dipetik Desember 20, 2020, from GNU Operating System: <https://www.gnu.org/philosophy/categories.html>
- Hatrix, V. (2019, February 4). *What is Rainmeter? How to Customize your Desktop with Rainmeter*. Dipetik Desember 20, 2021, dari www.viralhatrix.com: <https://www.viralhatrix.com/2019/02/what-is-rainmeter-how-to-customize-your.html>
- Pekkola, K. ' (2021, February 22). *What's on your desktop?* Dipetik Desember 1, 2020, dari [Rainmeter](http://www.rainmeter.net): <https://www.rainmeter.net/>
- Reddy, K. P. (2017). *Beginning Spring Boot Applications and Microservice with the Spring Framework*. Hyderabad, India: Apress.
- Sugiyono. (2018). *Metode Penelitian Kuantitatif, Kualitatif, dan R&D*. Jakarta: Alfabeta.
- Sulaeman1, F. S., & Permana, I. H. (2021). Sistem Monitoring Penerapan Rencana Anggaran Biaya Berbasis Web. *Jurnal IKRA-ITH TEKNOLOGI Vol 5 No 1*, 24-31.
- Sulasno, S., & Saleh, R. (2020). Desain dan Implementasi Sistem Monitoring Sumber Daya Server Menggunakan Zabbix 4.0. *JUITA: Jurnal Informatika e-ISSN: 2579-8901; Volume 8, Nomor 2, November 2020*, 1.
- Supriyatna, A. (2018). METODE EXTREME PROGRAMMING PADA PEMBANGUNAN WEB APLIKASI. *Universitas Bina Sarana Informatika*, 3.
- Sytsko, A. (2020, April 23). *Why Monitor Cpu Usage?* Dipetik Desember 20, 2020, dari Anturis [Blog](http://anturis.com/blog/why-monitor-): <https://anturis.com/blog/why-monitor->

cpu-usage/

- Tafonao, J., Fatkhiyah, E., & K, Y. R. (2019). PERANCANGAN DAN IMPLEMENTASI MONITORING SERVER OWN CLOUD. *Jurnal JARKOM Vol. 7 No. 2 Desember 2019*, 130-139.
- Tripathi, P., & Deshmukh, M. (2016). A Brief Introduction of Resource Management Techniques in Cloud Computing Environment. *international Journal on Recent and Innovation Trends in Computing and Communication Volume: 3 Issue: 6*, 4094 - 4099.
- Ulfi, M., Marthasari, G. I., & Nuryasin, I. (2020, 03 03). Implementasi Metode Personal Extreme Programming dalam Pengembangan Sistem Manajemen Transaksi Perusahaan. *CV. Todjoe Sinar Group*, 262.
- Vermaat, M. E., Sebok, S. L., Freund, S. M., Campbell, J. T., & Frydenberg, M. (2018). *Discovering Computers 2018*. Boston: Cengage Learning.
- Zehra, B. (2017). CPU-RAM bounded Processing for Video Game. *INTERNATIONAL JOURNAL OF COMPUTER SCIENCE AND EMERGING TECHNOLOGIES*, 49-52.