

CONTENT MANAGEMENT SYSTEM BERBASIS FIREBASE (STUDI KASUS WEBSITE LPPM STMIK PRIMAKARA)

I Made Ananda Widiprabawa¹, I Putu Satwika², Ketut Queena Fredlina³

¹Program Studi Sistem Informasi, STMIK Primakara

^{2,3}Program Studi Teknik Informatika, STMIK Primakara

Jalan Tukad Badung No. 135, Denpasar, Telp. (0361) 8956085

ananda.widiprabawa@gmail.com¹, satwika@primakara.ac.id², naa.queena@gmail.com³

Abstrak

Internet menjadi media yang informatif bagi perorangan maupun organisasi, salah satunya berupa *website*. Dengan menggunakan *website*, informasi sebuah perusahaan akan mudah didapat, dan nilai sebuah perusahaan akan meningkat. Untuk pengelolaan konten *website* diperlukan sebuah *Content Management System* (CMS). Salah satu CMS yang populer yaitu Wordpress. Tetapi memiliki keterbatasan untuk menggunakan fitur lebih seperti *plugin* berbayar. Dengan menggunakan Firebase, membuat *website* bisa tanpa biaya. Terdapat CMS Firebase yang bernama Flamelink. Sama seperti Wordpress, memiliki keterbatasan dalam penggunaannya. Dengan permasalahan tersebut dibuat CMS Firebase yang baru untuk mengatasi keterbatasan yang dimiliki Flamelink dan Wordpress. CMS Firebase ini akan diimplementasikan pada *website* LPPM STMIK Primakara. Hasil implementasinya yaitu pos dinamis, halaman statis dan dinamis, artikel penelitian, artikel pengabdian masyarakat, artikel dana hibah, agenda kegiatan, konten footer, konten statis untuk bagian jurnal, dan repositori jurnal dapat diimplementasikan.

Kata kunci: CMS, Firebase, Wordpress, Flamelink

Abstract

The internet has become an informative medium for individuals and organizations, one of which is a website. By using a website, a company's information will be easily obtained, and the value of a company will increase. For website content management a Content Management System (CMS) is needed. One popular CMS is Wordpress. But it has limitations to use more features such as paid plugins. By using Firebase, creating a website is free. There is a Firebase CMS called Flamelink. Just like Wordpress, has limitations in its use. With these problems a new Firebase CMS was created to overcome the limitations of Flamelink and Wordpress. This Firebase CMS will be implemented on the STMIK Primakara LPPM website. The results of implementation are dynamic posts, static and dynamic pages, research articles, community service articles, grant funding articles, activity agendas, footer content, static content for the journal section, and a journal repository can be implemented.

Keywords: CMS, Firebase, Wordpress, Flamelink

1. Pendahuluan

Dalam era industri 4.0 ini, internet berubah menjadi media yang informatif bagi siapapun baik perorangan maupun organisasi seperti perusahaan atau industri bidang kesehatan, keuangan, perbankan, hiburan, wisata, kuliner, pendidikan dan lainnya. Salah satu media internet yaitu *website*. Dengan menggunakan *website*, informasi sebuah perusahaan akan mudah didapat dan nilai sebuah perusahaan akan meningkat. Untuk memiliki sebuah *website* tanpa memerlukan biaya dan sumber daya yang besar yaitu dengan menggunakan *Content Management System* (CMS) dalam mengelola *website* perusahaan.

Salah satu contoh CMS yaitu Wordpress. Namun, untuk menggunakan konten yang benar-benar dinamis dan bisa disesuaikan harus menggunakan *plugin*, tetapi *plugin* ini terdapat

batasan untuk membuat konten dinamis dan harus membayar lebih untuk menggunakan *plugin*-nya secara penuh [1]. Wordpress masih menggunakan bahasa pemrograman PHP sebagai *backend*-nya dan MySQL sebagai *database*-nya. Untuk mempublikasikan *website* yang telah dibuat dengan Wordpress harus di-*hosting* dan berbayar, juga perlu melakukan konfigurasi CPanel. Jika ingin menggunakan *Content Delivery Network* (CDN), harus menggunakan pihak ketiga seperti Cloudflare. Beberapa model CMS seperti Flamelink telah tersedia. Akan tetapi terdapat banyak keterbatasan pada CMS Flamelink seperti user yang bisa menggunakan Flamelink terbatas hanya satu user, tidak ada *permission* atau *roles* pada *free plan*, dan tidak bisa disesuaikan karena tidak bersifat *open source*. Jika menginginkan fitur yang lebih maka harus membayar dengan harga yang lumayan mahal.

Firebase adalah sebuah layanan BaaS (Backend as a Service) dari Google. layanan ini menyediakan pengembang aplikasi API (Application Programming Interface) yang memungkinkan aplikasi data yang akan disinkronisasi di klien dan disimpan di cloud Firebase ini. Firebase menyediakan library untuk berbagai client platform yang memungkinkan integrasi dengan Android, iOS, JavaScript, Java, Objective-C dan Node.js dan dapat juga disebut sebagai layanan DbaaS (Database as a Service) dengan konsep realtime. Firebase digunakan untuk mempermudah dalam penambahan fitur-fitur yang akan dibangun oleh developer [2]. Susanti dan Triyono [3] telah mengembangkan teknologi web sistem pemantau kendaraan secara realtime dengan menggunakan teknologi AngularFire (AngularJS dan Firebase). Pada ujicoba tersebut, sistem pemantau adalah dengan melakukan akuisisi data posisi koordinat dari satelit GPS pada kendaraan, mengirimkan ke server Firebase dan selanjutnya menampilkannya secara real time melalui web dengan visualisasi Google Maps. Nurzam dkk [4] mengembangkan sistem Cloud Messaging berbasis Mobile menggunakan sistem Firebase. Pada ujicoba tersebut, implementasi sistem Firebase dapat menghasilkan aplikasi sistem layanan yang dinamis berupa fitur push notification Firebase Cloud Messaging untuk mengirimkan pesan notifikasi lintas perangkat pada berbagai aplikasi lain sesuai dengan kebutuhan.

Artikel ini menyajikan konsep CMS berbasis Firebase yang dapat membuat tipe konten secara dinamis seperti konten blog, *event*, lokasi, *media partner*, dan lainnya. CMS ini diimplementasikan pada *website* LPPM STMIK Primakara karena *website* sebelumnya terkena *hijacked* dan belum diperbaiki, serta untuk mengetahui CMS ini dapat mengakomodir konten pada *website* Lembaga Penelitian dan Pengabdian Masyarakat (LPPM) STMIK Primakara.

2. Tinjauan Pustaka

Siambaton [5] meneliti mengenai Aplikasi Content Management System pada Joomla untuk membuat Web Service menggunakan satu jenis konten. Hasil penelitian ini yaitu fitur CMS yang diimplementasikan pada aplikasi yang dibangun antara lain pengelolaan konten, pengelolaan kategori konten, pengelolaan user, dan penampilan konten pada halaman web.

Ulinuha dan Irwan [6] meneliti mengenai *Content Management System* untuk Tes *Online* Mahasiswa Baru pada Universitas Muria Kudus menggunakan bahasa pemrograman PHP dan *database* MySQL, serta metode Waterfall untuk pengembangan. Hasil penelitian ini adalah situs tes *online* yang menitikberatkan pada manajemen konten dan penambahan pengelolaan tes.

Sasongko [7] meneliti mengenai Perancangan *Content Management System* Weblog Organisasi menggunakan Pendekatan Teknik *Model View Controller* (MVC). Hasil penelitian ini yaitu CMS yang telah dibangun secara umum dapat menyajikan informasi, baik berupa teks, gambar, dan konten HTML bagi yang memahami pemrograman web.

Padmawati [8] meneliti mengenai Perancangan Content Management System Program Studi Universitas Dhyana Pura menggunakan Metode *Task Centered System Design* (TCSD). Pengujian sistem menggunakan BlackBox dan dari hasil pengujian, tujuan dari pembuatan CMS ini telah tercapai.

Sentosa [9] mengembangkan Web Konten Manajemen Sistem Secara Dinamis dengan Bahasa Pemrograman PHP *Framework* CodeIgniter dengan *Database* MariaDB. Hasil dari penelitian ini yaitu CMS yang dibuat dapat melakukan penyebaran berita dengan baik dan pengaksesan halaman *website* yang cepat dan ringan.

Shah [10] meneliti mengenai *Building A Web Content Management System* dengan hasil penelitian yaitu CMS yang diimplementasikan pada bagian *News and Events* pada

Computer Science Department, San Diego State University dengan fitur kemudahan mengelola konten, terdapat *templates*, *user authentication*, dan *workflow management*.

Rachmaniah dan Ghifari [11] mengembangkan Web e-commerce menggunakan CMS Drupal. Penelitian ini menggunakan pendekatan prototyping dan mengimplementasikan 193 modul dari 317 modul relevan yang tersedia pada CMS Drupal Kickstart2. Hasil penelitian menunjukkan sistem mampu mengintegrasikan modul-modul yang tersedia pada CMS Drupal Kickstart 2 sehingga e-Commerce yang dihasilkan bersifat interaktif. CMS Drupal mudah untuk diinstal dan bersifat *open source*, namun dengan harga yang mahal dan tidak didesain untuk menghasilkan tampilan yang *user friendly*.

Paper ini mengajukan konsep penggunaan CMS berbasis Firebase yang dapat digunakan untuk mengembangkan berbagai jenis konten yang dinamis yang disesuaikan dengan kebutuhan.

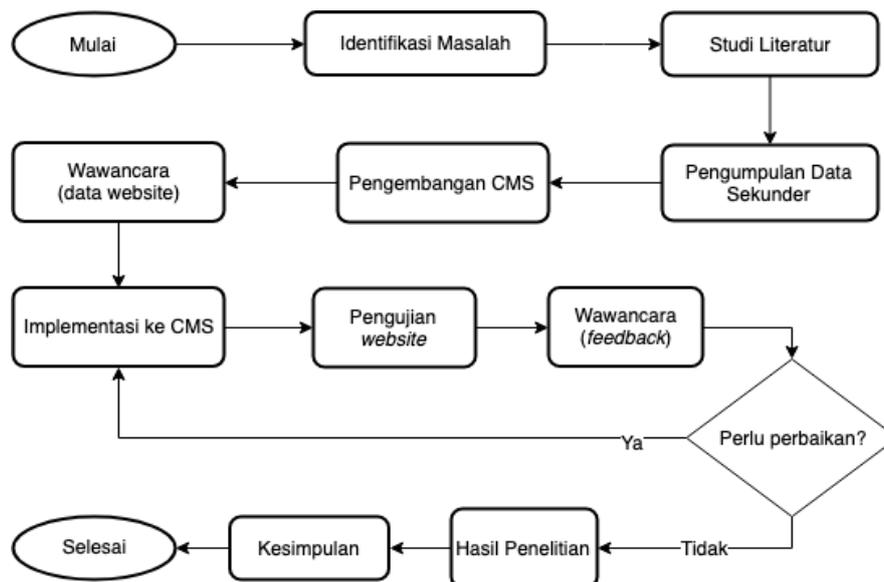
3. Metodologi

Jenis penelitian ini yaitu pengembangan dengan metode pengembangan CMS yang digunakan adalah metode Agile yang dimodifikasi. *Agile* merupakan sebuah metode yang proses pengembangannya dilakukan secara iteratif, dimana persyaratan atau kebutuhan bisa berubah sewaktu-waktu. Metode Agile ini dimodifikasi karena dalam mengembangkan CMS dengan Firebase akan terjadi perubahan struktur database menyesuaikan terhadap fitur yang dibuat, dan perubahan atau penambahan fitur yang tidak menentu. Tahapan pengembangan yang dilakukan yaitu sebagai berikut.

- 1) *Plan*
Pada tahap ini membuat rencana atau definisi kebutuhan CMS yang dibuat yaitu fitur *content type*, *pages*, *authentication*, *media file image*, *theme setting*, dan *site setting*. Serta rencana perbaikan ketika *bug* ditemukan setelah dilakukan *test*.
- 2) *Development*
Setelah perencanaan dilanjutkan dengan pengembangan CMS berdasarkan kebutuhan yang sudah didefinisi pada tahap sebelumnya.
- 3) *Testing*
Fitur yang sudah dikembangkan diuji coba untuk mengetahui fungsinya sudah bekerja dengan baik.
- 4) *Deployment*
Jika fitur yang sudah diuji coba berfungsi dengan baik, maka CMS di-*update* ke versi terbaru, jika tidak maka kembali ke tahap perencanaan untuk perbaikan.

Teknik yang digunakan untuk mengumpulkan data dalam penelitian ini yaitu studi literatur dan wawancara. Studi literatur yang digunakan yaitu jurnal-jurnal yang berkaitan dengan CMS dan Firebase. Studi literatur bertujuan untuk mengumpulkan fitur-fitur CMS yang nantinya akan diimplementasikan pada CMS yang dikembangkan. Dan wawancara yang digunakan merupakan wawancara tidak terstruktur atau secara mendalam yang berarti wawancara yang bebas dimana peneliti tidak menggunakan pedoman yang tersusun secara sistematis untuk pengumpulan datanya karena pertanyaan nantinya berdasarkan pada data dan *feedback* yang didapat selama wawancara. Wawancara dalam penelitian ini dilakukan terhadap individu yang mengetahui tentang kriteria, data dan informasi *website* LPPM yaitu Ketua LPPM STMIK Primakara. Wawancara ini bertujuan untuk mendapatkan data dan kriteria *website* LPPM STMIK Primakara. Alat bantu yang digunakan untuk mengumpulkan data penelitian yaitu *smartphone* dengan aplikasi *voice recorder* untuk merekam percakapan selama wawancara berlangsung.

Adapun data yang digunakan dalam penelitian ini adalah kualitatif. Data kualitatif bersifat deskriptif dimana data yang terkumpul berbentuk kata-kata atau gambar, sehingga tidak menekankan pada angka. Data yang dikumpulkan dari penelitian ini berasal dari dua sumber, yaitu data primer, yang diperoleh langsung dari hasil wawancara dengan Ketua LPPM. Kedua yaitu data sekunder, yang diperoleh dari internet dan jurnal.



Gambar 1. Alur Penelitian

Langkah-langkah pada gambar 1 dapat dijelaskan sebagai berikut.

- 1) Pertama, dimulai dengan identifikasi masalah.
- 2) Kemudian mencari referensi dengan melakukan studi literatur sebagai acuan penelitian.
- 3) Mengumpulkan data sekunder berupa fitur-fitur standar pada CMS yang diperoleh dari hasil melakukan studi literatur.
- 4) Setelah data sekunder terkumpul, dilanjutkan dengan mengembangkan CMS.
- 5) Melakukan wawancara pertama untuk mendapatkan data dan kriteria website LPPM kepada Ketua LPPM.
- 6) Membuat website menggunakan CMS berdasarkan data dan kriteria yang diperoleh sebelumnya.
- 7) Website yang telah dibuat diujicobakan menggunakan Lighthouse untuk mengetahui kinerja website.
- 8) Dilanjutkan dengan wawancara kedua untuk mendapatkan *feedback* terhadap website sambil memperlihatkan website dan kinerjanya kepada ketua LPPM.
- 9) Jika terdapat perbaikan maka dilanjutkan lagi dengan mengimplementasikan perbaikan tersebut pada CMS.
- 10) Jika tidak, maka dilanjutkan dengan membuat hasil penelitian.
- 11) Berdasarkan hasil penelitian, maka diperoleh kesimpulan.

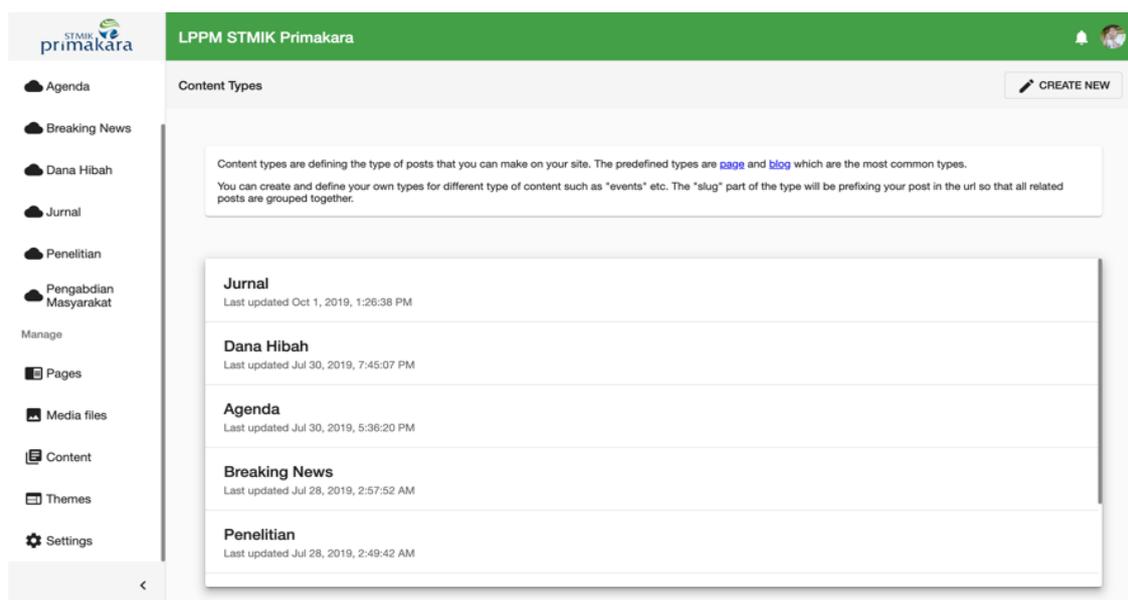
4. Hasil dan Pembahasan

4.1 Fitur CMS yang Dapat Mengakomodir Website LPPM STMIK Primakara

Berdasarkan hasil wawancara, kriteria *website* yaitu pos dinamis, halaman statis, artikel penelitian, artikel pengabdian masyarakat, artikel dana hibah, agenda kegiatan, kalender kegiatan, repositori jurnal, arsip agenda, link jurnal, dan konten *footer*. Kriteria yang dapat diakomodir oleh fitur CMS antara lain, pos dinamis, artikel penelitian, artikel pengabdian masyarakat, artikel dana hibah, jurnal, dan agenda kegiatan diakomodir oleh fitur tipe konten (*content type*). Halaman statis dan konten footer diakomodir oleh fitur halaman (*page*). Link jurnal diakomodir oleh fitur *template*. Sedangkan arsip agenda tidak dapat diakomodir karena *template* CMS tidak dapat melakukan pengecekan waktu. Dan kalender kegiatan juga tidak dapat diakomodir karena CMS tidak bisa memunculkan konten pada kalender.

a) Tipe Konten (*Content Type*)

Konten atau pos bisa dibuat secara dinamis beserta *field* yang bisa disesuaikan dengan kebutuhan melalui fitur tipe konten. Konten yang dibuat berdasarkan hasil wawancara yaitu artikel penelitian, pengabdian masyarakat, dana hibah, jurnal dan agenda kegiatan. Tampilan tipe konten pada CMS dapat dilihat pada gambar 2.



Gambar 2. Tampilan *Content Type* pada CMS

b) Halaman (*Pages*)

Pembuatan halaman statis dan halaman dinamis diatur dalam menu *Pages*. Pada kasus ini dibuat halaman *About* sebagai halaman statisnya. Untuk memunculkan halaman pada *website* perlu melakukan pengaturan *routing* pada *template theme*.

c) Template

Membuat tampilan secara *custom* dilakukan melalui fitur template ini. Tetapi dalam pembuatannya diperlukan pengetahuan HTML dan CSS. Semua tampilan *website* dibuat dalam template ini, dan tampilan *custom* berdasarkan kriteria *website* yang diakomodir yaitu link jurnal dan konten *footer*.

4.2 Hasil Implementasi CMS pada Website LPPM STMIK Primakara

Dari hasil *feedback* wawancara, ada fitur atau kebutuhan yang tidak dapat diakomodir dengan menggunakan CMS ini yaitu, arsip agenda karena *template* CMS tidak dapat membandingkan waktu ataupun tanggal. Maka dari itu, arsip agenda ditampilkan secara keseluruhan tanpa berdasarkan waktu atau tanggal. Dan kalender kegiatan ditampilkan menggunakan alternatif Google Calendar.

a) Halaman Beranda

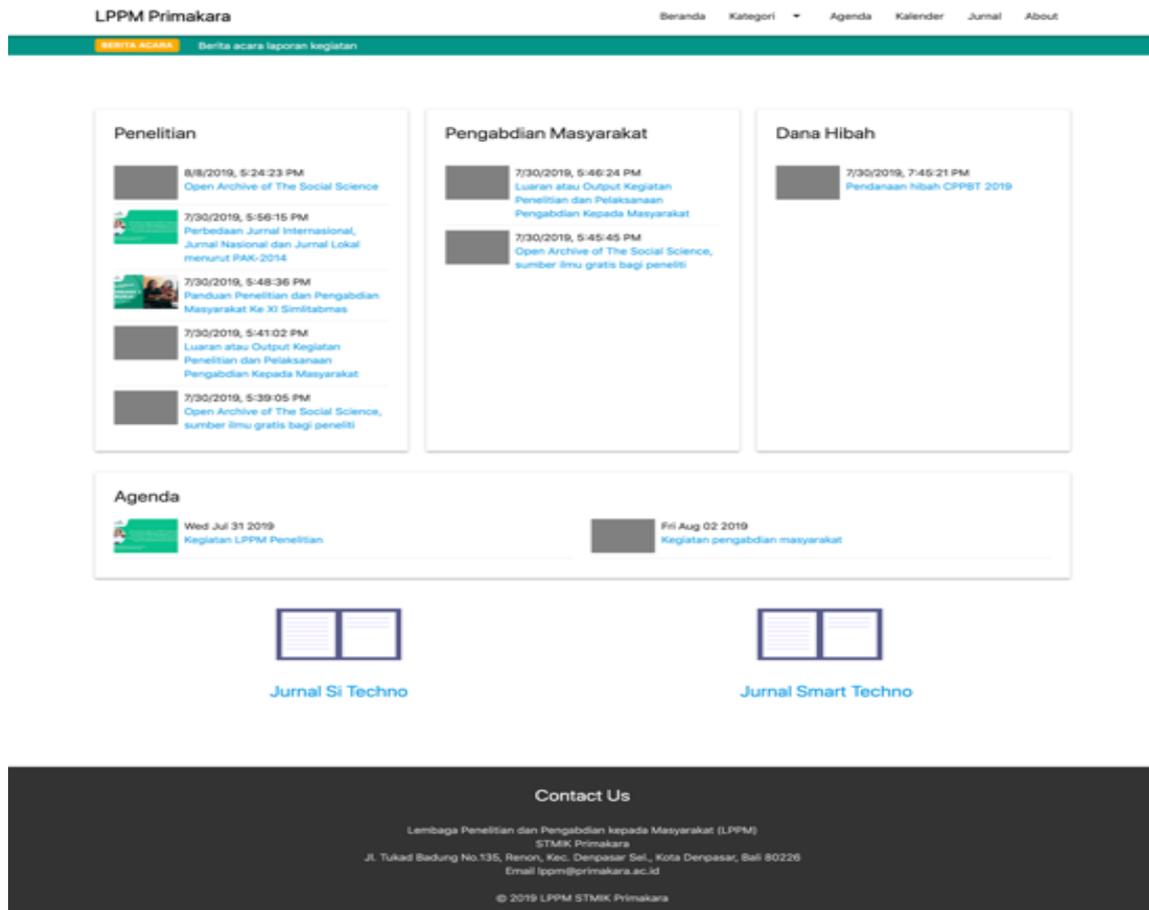
Konten yang tampil pada halaman beranda berupa navigasi menu, artikel penelitian, artikel pengabdian masyarakat, artikel dana hibah, agenda kegiatan, bagian jurnal, dan *footer*, seperti terlihat pada gambar 3.

b) Halaman Artikel

Terdapat dua jenis halaman artikel yaitu, halaman artikel yang memuat semua satu topik artikel dan yang kedua yaitu halaman yang menampilkan satu artikel secara lengkap. Semua halaman artikel dan *detail* halaman artikel memiliki *template* yang sama. Terkecuali halaman *detail* agenda memiliki tampilan yang berbeda dan tambahan konten tanggal kegiatan.

c) Halaman Statis

Terdapat satu halaman statis yang diimplementasikan pada *website* yaitu halaman *about*. Dibuat menggunakan fitur *pages* pada CMS. Halaman yang sudah dibuat harus diatur *routing*-nya pada bagian *template theme* supaya halaman tersebut bisa diakses.



Gambar 3. Halaman Beranda Website



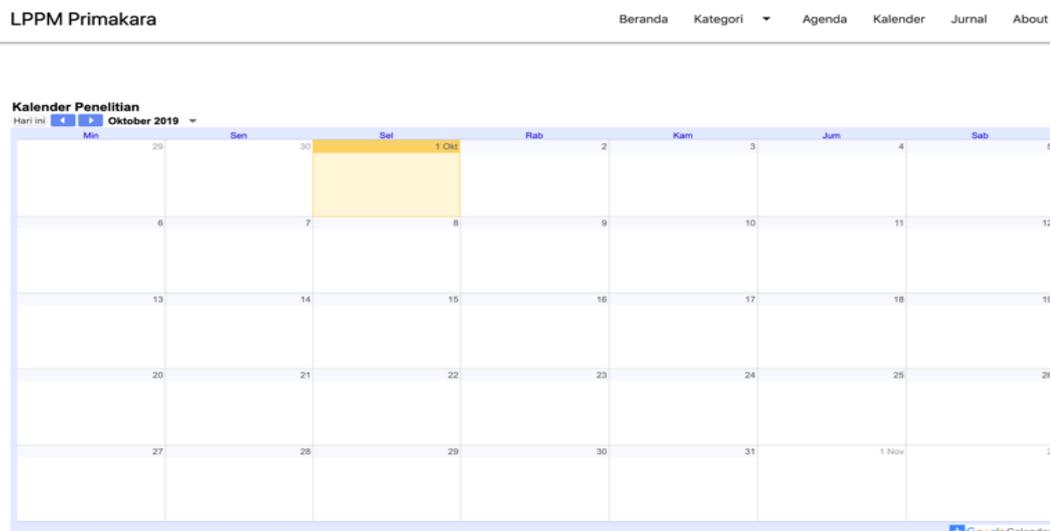
Gambar 4. Halaman Statis About

d) Konten Footer

Konten pada footer dibuat langsung melalui *template* dan untuk mengatur tata letak dan warna menggunakan HTML dan CSS.

e) Google Calendar

Pada CMS ini tidak dapat membuat kalender karena keterbatasan untuk mengambil data pada sisi *client side*. Sebagai alternatifnya maka digunakan Google Calendar untuk menampilkan konten kegiatan penelitian. Cara mengintegrasikan Google Calendar ini yaitu dengan meng-*embed* kode yang didapat melalui halaman Google Calendar kedalam *template* CMS.



Gambar 5. Google Calendar

f) Repositori Jurnal

Repositori jurnal dibuat melalui *content type* dan ditampilkan dalam bentuk daftar tabel pada halaman tersendiri.

No	Nama Jurnal	Penerbit	Bidang	Terakreditasi Dikti	Terindex	Terbit
1	Jurnal Pendidikan IPA Indonesia (Indonesian Journal of Science Education)	Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Negeri Semarang	Pendidikan	Google	Scopus	2019-10-01
2	Jurnal Cakrawala Pendidikan	Lembaga Pengembangan dan Penjaminan Mutu Pendidikan UNY	Pendidikan	Google	Scopus	2019-10-01

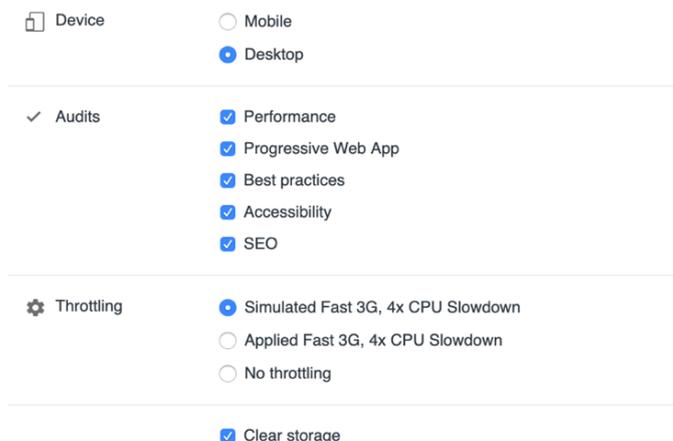
Contact Us

Lembaga Penelitian dan Pengabdian kepada Masyarakat (LPPM)
 STMIK Primakara
 Jl. Tukad Badung No.135, Renon, Kec. Denpasar Sel., Kota Denpasar, Bali 80226
 Email: lppm@primakara.ac.id

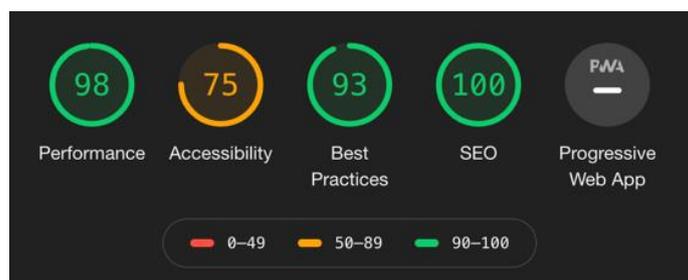
Gambar 6. Repositori Jurnal

g) Audit *Website*

Audit kinerja *website* dan SEO dilakukan menggunakan alat Lighthouse pada browser Google Chrome dengan pengaturan seperti gambar 7. Setelah diuji coba menggunakan Lighthouse didapatkan hasil kinerja *website* 98/100, aksesibilitas 75/100, *best practice* 93/100 dan SEO 100/100 seperti yang terlihat pada gambar 8



Gambar 7. Pengaturan Lighthouse



Gambar 8. Hasil audit Lighthouse

Terdapat tiga skala penilaian yaitu nilai 0-49 (rendah), 50-89 (sedang), dan 90-100 (tinggi). Dan berikut merupakan panduan penilaian masing-masing kategori.

1. *Performance Score*

Skor kinerja antara 0 sampai dengan 100. Skor 0 menunjukkan terdapat *error* pada Lighthouse dan skor 100 menunjukkan kinerja terbaik. Bagian *metrics* pada kinerja mempengaruhi skor, sedangkan bagian *diagnostics* dan *opportunities* tidak berpengaruh terhadap skor kinerja tetapi bias membantu untuk membuat halaman dimuat lebih cepat.

2. *Progressive Web App (PWA) Score*

Audit PWA berdasarkan pada *Baseline PWA checklist* yang terdiri dari 14 daftar kebutuhan.

3. *Accessibility Score*

Setiap 1 daftar audit aksesibilitas bisa berhasil ataupun gagal, tidak seperti kinerja yang memiliki *range* penilaian untuk setiap daftarannya.

4. *Best Practices Score*

Untuk menghitung seberapa banyak setiap audit berpengaruh terhadap skor secara keseluruhan, hitung jumlah audit kemudian dibagi dengan 100.

5. Kesimpulan

Berdasarkan hasil penelitian yang dilakukan, dapat disimpulkan bahwa: 1) Fitur yang dapat mengakomodir *website* LPPM STMIK Primakara yaitu tipe konten (*content type*), halaman (*pages*), dan *templates*. 2) Hasil implementasi CMS pada *website* LPPM STMIK Primakara yaitu pos dinamis, halaman statis dan dinamis, artikel penelitian, artikel pengabdian masyarakat, artikel dana hibah, agenda kegiatan, konten footer, konten statis untuk bagian jurnal yang mengarah ke OJS, dan repositori jurnal dapat diimplementasikan, kalender kegiatan menggunakan alternative Google Calendar dan yang tidak dapat diimplementasikan pada CMS ini yaitu arsip agenda.

DAFTAR REFERENSI

- [1] Agatha N. Tutorial Wordpress – 7 Plugin Contact Form Wordpress Terbaik. <https://www.hostinger.co.id/tutorial/contact-form-wordpress/>, 29 Juli, 2019: 15.20.
- [2] Abdurahman, D., & Kurnia, M. A. Rancang Bangun Aplikasi Pelayanan Donor Darah Pada PMI Kabupaten Majalengka Berbasis Android Dan Web Menggunakan Teknologi Firebase. *INFOTECH journal*, 2019; 5(1): 53-59.
- [3] Susanti, E., Triyono, J., & Pi, R. Pengembangan sistem pemantau dan pengendali kendaraan menggunakan raspberry pi dan firebase. *Jurnal Informatika*. 2016; 1: 144-153.
- [4] Nurzam, F., Fajri, I. N., & Prabowo, D. Rancang Bangun Aplikasi Media Laporan Aspirasi Dengan Firebase Cloud Messaging Berbasis Mobile. *SEMNAS TEKNO MEDIA ONLINE*. 2017; 5(1), 4-5.
- [5] Siambaton MZ, Fakhriza M. Aplikasi Content Management System (CMS) Pada Joomla Untuk Membuat Web Service. *InfoTekJar (Jurnal Nasional Informatika dan Teknologi Jaringan)*. 2016;1(1):11–23.
- [6] Ulinuha A, Irawan Y. Content Management System (CMS) Untuk Tes Online Mahasiswa Baru Pada Universitas Muria Kudus. *Jurnal SIMETRIS*. 2014;4(1):11.
- [7] Sasongko A. Perancangan Content Management System (CMS) Weblog Organisasi Menggunakan Pendekatan Teknik Model View Controller (MVC). *Jurnal Khatulistiwa Informatika*. 2013; 1(4):7–16.
- [8] Padmawati NPT, Feoh G, Gunawan PW. Perancangan Content Management System Program Studi Universitas Dhyana Pura Menggunakan Metode TCSD. *Jurnal Teknologi Informasi dan Komputer*. 2017; 2(1): 313–24.
- [9] Sentosa RB. Membangun Web Konten Manajemen Sistem Secara Dinamis Dengan Bahasa Pemrograman PHP Framework CodeIgniter Dengan Database MariaDB. *Journal of Information Technology and Computer Science (INTERCOMS)*. 2018; 1(2): 12–23.
- [10] Shah, R. V. *Building a web content management system*, Doctoral dissertation. San Diego: San Diego State University, 2012.
- [11] Rachmaniah, M., & Ghifari, R. A. (2019). E-Commerce B2B Getah Pinus Menggunakan CMS Drupal Kickstart 2. *Jurnal Ilmu Komputer dan Agri-Informatika*. 2019; 5(2), 88-98.