p-ISSN : 2406-7733
e-ISSN : 2597-9922

# RANCANG BANGUN SISTEM INFORMASI MANAJEMEN PUBLIKASI BERITA PARAGRAF MUDA PADA UKM JURNALISTIK MENGGUNAKAN *AGILE SCRUM*

**Cahyani Budihartanti[1], Alfian Eka Maulana[2], Muhammad Farhan Nur Hidayat[3], Muhammad Luthfi Pratama[4]**
Program Studi Sistem Informasi, Fakultas Teknologi Informasi, Universitas Nusa Mandiri[1]
Program Studi Sistem Informasi, Fakultas Teknik dan Informatika, Universitas Bina Sarana Informatika[2, 3, 4]
E-mail: cahyani.cbh@nusamandiri.ac.id[1], alfianmaulana114@gmail.com[2], 04102002farhan@gmail.com[3], luthfipratama1012@gmail.com[4]

**Abstrak** - Unit Kegiatan Mahasiswa (UKM) di era digital sering kali menghadapi kendala dalam mengelola data operasional dan konten karena masih menggunakan *platform* yang terpisah, seperti penggunaan Blogspot sebagai portal berita dan *spreadsheet* untuk pencatatan operasional. Pemisahan ini menyebabkan data menjadi tersebar, sulit dilacak riwayatnya, rawan tidak konsisten, dan menyulitkan penyusunan laporan. Penelitian ini bertujuan untuk merancang sistem informasi manajemen berbasis web bernama "Paragraf Muda" yang mengintegrasikan pengelolaan konten jurnalistik dan data operasional UKM secara terpusat. Pengembangan perangkat lunak dilakukan menggunakan metode *Agile* dengan kerangka kerja *Scrum*, guna mengakomodasi kebutuhan sistem UKM yang dinamis secara bertahap, iteratif, dan adaptif terhadap masukan pengguna. Secara teknis, sistem dimodelkan dengan *Unified Modeling Language* (UML) dan *Entity Relationship Diagram* (ERD), serta diimplementasikan menggunakan bahasa pemrograman PHP melalui framework Laravel dan database MySQL. Hasil penelitian menunjukkan bahwa pendekatan Scrum yang dibagi ke dalam beberapa tahapan *sprint* mampu memfasilitasi pembangunan fitur secara efektif. Sistem yang dihasilkan berhasil mengimplementasikan role-based access control untuk memisahkan ruang kerja dan wewenang secara jelas, sehingga alur kerja redaksional dari pembuatan *brief* oleh divisi Litbang, penulisan draf oleh anggota Redaksi, penyuntingan oleh Kepala Redaksi, hingga persetujuan dan publikasi akhir oleh Koordinator dan divisi Humas dapat berjalan secara terstruktur dan efisien.

**Kata Kunci:** *Agile Scrum*, Laravel, Perangkat Lunak Berbasis Web, Sistem Informasi Manajemen, UKM Jurnalistik.

## I. PENDAHULUAN

Unit Kegiatan Mahasiswa (UKM) adalah wadah bagi mahasiswa dengan minat yang sama dan berperan penting dalam mengembangkan kemampuan di luar perkuliahan, termasuk dalam hal informasi dan komunikasi kampus (Hutauruk & Pakpahan, 2021).Di era digital, UKM jurnalistik tidak cukup hanya aktif secara luring, tetapi juga memerlukan sistem yang rapi untuk mengelola konten dan data organisasi.

Saat ini, UKM jurnalistik yang menjadi objek pengembangan masih menggunakan Blogspot sebagai portal berita dan *spreadsheet* untuk mengelola data anggota, absensi, notulensi, jadwal, dan keuangan. Pemisahan ini membuat data tersebar di banyak file, sulit dilacak riwayatnya, rawan tidak konsisten, dan menyulitkan penyusunan laporan serta pemantauan keaktifan anggota.

Penelitian (Hutauruk & Pakpahan, 2021) menunjukkan bahwa organisasi kemahasiswaan menghadapi masalah serupa karena proses administrasi dan penyebaran informasi masih banyak dilakukan secara manual, sehingga dibutuhkan sistem informasi berbasis web untuk membantu pengelolaan organisasi. Penelitian lain tentang sistem absensi menegaskan bahwa pencatatan manual sering memicu kesalahan dan keterlambatan informasi, sehingga absensi digital berbasis web sangat diperlukan untuk meningkatkan kecepatan, efisiensi, dan akurasi pengelolaan data kehadiran (Nasution et al., 2024). Di sisi pengembangan perangkat lunak, metode *Agile*, khususnya *Scrum*, banyak digunakan karena bersifat iteratif dan fleksibel terhadap perubahan kebutuhan pengguna. Untuk mendukung fleksibilitas tersebut, penggunaan *framework* Laravel berbasis PHP dan *database* MySQL menjadi solusi teknis yang relevan karena kemampuannya dalam mengelola data yang terintegrasi secara aman(Trisanto et al., 2023).

Berdasarkan latar belakang tersebut, rumusan masalah dalam pengembangan sistem ini adalah: bagaimana merancang sistem informasi berbasis web menggunakan metode *Agile/Scrum* yang dapat mengintegrasikan pengelolaan konten jurnalistik dan data operasional UKM; bagaimana mengimplementasikan fitur-fitur utama seperti manajemen berita, konten media sosial, keanggotaan, absensi, jadwal kegiatan, dan keuangan dalam satu sistem terpusat; serta bagaimana menerapkan metode *Agile* dengan kerangka kerja *Scrum* agar pengembangan sistem lebih adaptif terhadap kebutuhan pengguna.

p-ISSN : 2406-7733
e-ISSN : 2597-9922

Tujuan pengembangan sistem ini adalah merancang sistem informasi berbasis web yang mengintegrasikan pengelolaan konten jurnalistik dan data operasional UKM jurnalistik yang sebelumnya dikelola secara terpisah, mengimplementasikan fitur-fitur utama dalam satu platform terpusat, serta menerapkan metode Agile dengan kerangka kerja Scrum mulai dari perencanaan *backlog*, pelaksanaan *sprint*, hingga evaluasi dan perbaikan berulang.

Ruang lingkup laporan ini dibatasi pada pengembangan sistem informasi berbasis web untuk satu UKM jurnalistik di lingkungan kampus. Secara teknis, pengembangan dilakukan menggunakan *framework* Laravel 12 dengan bahasa pemrograman PHP dan MySQL sebagai pengelola basis data. Sistem difokuskan pada pengelolaan berita, konten media sosial, data anggota, absensi, jadwal kegiatan, dan catatan keuangan sederhana tanpa integrasi dengan sistem kampus lain atau layanan pihak ketiga. Proses pengembangan dibahas dalam konteks penerapan metode *Agile/Scrum* pada skala proyek mata kuliah, dan pengujian dilakukan secara sederhana dengan melibatkan beberapa pengguna UKM jurnalistik untuk memastikan fitur utama dapat digunakan.

## II. TINJAUAN PUSTAKA

### 2.1 *Agile/Scrum*

Pendekatan *Agile* dalam pengembangan perangkat lunak dilakukan secara iteratif dan bertahap dengan memprioritaskan fleksibilitas, sinergi tim, serta ketangkasan dalam merespons dinamika kebutuhan pengguna. Guna menyelaraskan prinsip tersebut, penggunaan kerangka kerja *Scrum* membagi proses adaptasi ini ke dalam siklus jangka pendek yang dikenal sebagai *sprint*, sehingga seluruh tahapan pengembangan sistem menjadi lebih terstruktur dan optimal (Irfansyah et al., 2026).

### 2.2 Organisasi Kemahasiswaan (UKM)

Organisasi kemahasiswaan adalah perkumpulan, kesatuan mahasiswa yang sudah terlembaga, mempunyai tujuan jelas guna mengembangkan peran serta dan fungsi mahasiswa (Avianti et al., 2024).

### 2.3 Sistem Informasi Manajemen

Sistem Informasi Manajemen (SIM) adalah suatu alat (berupa system informasi) untuk menghasilkan informasi (yang berkualitas) guna mendukung manajemen dalam proses pengambilan keputusan (Patma et al., 2018).

### 2.4 Laravel

Laravel adalah salah satu framework PHP terbaik yang dikembangkan oleh Taylor Otwell, proyek Laravel dimulai pada April 2011. Awal mula, proyek ini dibuat, karena Otwell sendiri tidak menemukan framework yang up-to-date dengan versi PHP. Mengembangkan *framework* yang sudah ada juga bukan merupakan ide yang bagus karena keterbatasan sumber daya (Yudhanto & Prasetyo, 2019).

### 2.5 PHP

PHP adalah singkatan rekursif dari PHP: *Hypertext Preprocessor*. Ini adalah bahasa pemrograman *open-source* yang sangat populer, dirancang pengembangan web. PHP adalah fondasi utama dari banyak situs web terbesar di dunia termasuk WordPress, Facebook di masa awal (Sophan et al., 2025).

### 2.6 MySQL

MySQL adalah DBMS yang open source dengan dua bentuk lisensi, yaitu *Free Software* (perangkat lunak bebas) dan *Shareware* (perangkat lunak berpemilik yang penggunaannya terbatas). Jadi, MySQL adalah database server yang gratis dengan lisensi GNU *General Public License* (GPL) sehingga dapat dipakai untuk keperluan pribadi atau komersil tanpa harus membayar lisensi yang ada (Namruddin et al., 2023).

### 2.7 *Produck Backlog*

*Product Backlog* adalah daftar prioritas yang berisi semua kebutuhan, fitur, perbaikan, dan perubahan yang diinginkan untuk produk yang sedang dikembangkan. Ini merupakan tanggung jawab *Product Owner* untuk mengelola Product Backlog dan memastikan item-item di dalamnya diperbarui dan diprioritaskan dengan benar (Magdalena, 2023).

### 2.8 UML

*Unified Modeling Language* (UML) adalah sebuah standar dalam perancangan perangkat lunak yang berorientasi pada objek, dan digunakan untuk memodelkan perangkat lunak yang akan dibangun dalam bentuk dokumen (Sihombing et al., 2025).

### 2.9 ERD

ERD adalah gambar atau diagram yang menunjukkan informasi dibaat, disimpan, dan digunakan dalam sistem bisnis. Entitas biasanya menggambarkan jenis informasi yang sama. Dalam entitas digunakan untuk menghubungkan antar entitas yang sekaligus menunjukkan hubungan antar data (Suprapto, 2021).

### 2.10 LRS

*Logical Relationship Schema* (LRS) adalah representasi yang lebih rinci dari struktur basis data, termasuk tabel-tabel, kolom-kolom. kunci-kunci, dan hubungan antar tabel. LRS adalah langkah

p-ISSN : 2406-7733
e-ISSN : 2597-9922

selanjutnya setelah ERD dalam merancang basis data, dan biasanya diwakili dalam bentuk skema basis data yang terstruktur(Romadloni & Miswanto, 2024).

### 2.11 *Sprint*

Setelah cara dan batas waktu pengerjaan ditentukan dalam *sprint planning*, tahapan metode scrum selanjutnya adalah melakukan *sprint*. Proses ini dilakukan melalui kegiatan daily scrum (Magdalena, 2023).

### 2.12 *Blacbox Testing*

*Blackbox Testing* merupakan salah satu metode yang biasa digunakan untuk melakukan pengujian sistem, pengujian ini menguji apakah keluaran dari suatu fungsi sesuai dengan *Test Case* yang telah dibuat Pengujian kotak hitam adalah pengujian yang dilakukan berdasarkan spesifikasi persyaratan dan tidak melibatkan pemeriksaan kode aplikasi (Winarti & Albaqir, 2025).

### 2.13 UAT

*User Acceptance Testing* (UAT) adalah pengujian yang bertujuan untuk memastikan bahwa aplikasi mampu memenuhi komponen komponen dalam dokumen bisnis serta mampu diterima oleh *user*. Pengujian ini perlu mendapatkan *feedback* dari *user* dengan Level N-tujuan apakah aplikasi mudah digunakan dan mampu memenuhi kebutuhan bisnis perusahaan atau tidak (Diyasa et al., 2025).

## III. METODE PENELITIAN

### 3.1 Metode Penelitian

Penelitian ini menggunakan metode *Agile* dengan kerangka kerja Scrum sebagai pendekatan pengembangan sistem. Metode ini dipilih karena kebutuhan di UKM jurnalistik sifatnya cukup dinamis dan sering kali baru terlihat jelas setelah pengurus mencoba langsung prototipe sistem yang dibuat. Dengan *Scrum*, pengembangan bisa dilakukan secara bertahap sambil terus menyesuaikan diri dengan masukan dari pengguna.

Pengembangan dimulai dari penyusunan product backlog berdasarkan hasil diskusi dan peninjauan dokumen yang digunakan pengurus UKM Jurnalistik Paragraf Muda dalam aktivitas sehari-hari. Daftar tersebut kemudian diolah dalam *sprint* planning untuk menentukan fitur dan tugas yang akan dikerjakan pada setiap iterasi pengembangan sesuai prioritas kebutuhan organisasi.

Pada tahap *sprint*, sistem dikembangkan secara bertahap menggunakan Laravel 12 dan MySQL. Setiap modul diimplementasikan, diuji secara internal, dan diintegrasikan dengan modul yang sudah ada sehingga pada akhir *sprint* selalu tersedia versi sistem yang bisa dicoba. Setelah itu dilakukan *sprint* review bersama pengurus UKM untuk mendemokan fitur yang sudah berjalan dan mengumpulkan masukan. Tahap ini diikuti dengan *sprint retrospective*, di mana tim mengevaluasi proses kerja selama *sprint* dan menentukan perbaikan yang perlu dilakukan pada iterasi berikutnya. Dengan pola ini, pengembangan sistem bisa berjalan lebih terkontrol, namun tetap fleksibel terhadap perubahan kebutuhan pengguna.

### 3.2 Alat Bantu Perancangan Sistem

Perancangan sistem Paragraf Muda diawali dengan pemodelan agar alur dan struktur sistem jelas sebelum diimplementasikan. *Unified Modeling Language* (UML) digunakan untuk menggambarkan kebutuhan dan perilaku sistem, melalui *use case* diagram, *activity* diagram, *sequence* diagram, serta *component* dan *deployment* diagram yang menjelaskan aktor, alur proses, interaksi komponen, dan lingkungan sistem.

Pada sisi basis data, rancangan dibuat menggunakan *Entity Relationship Diagram* (ERD) untuk menentukan entitas dan relasinya, kemudian dijabarkan ke dalam *Logical Record Structure* (LRS) sebagai acuan pembentukan tabel di MySQL. Dengan rancangan ini, struktur data tetap konsisten dan mendukung pengembangan modul-modul pada setiap *sprint*.

### 3.3 Teknik Pengumpulan Data

Data penelitian diperoleh melalui observasi dan studi pustaka. Observasi dilakukan pada aktivitas pengelolaan berita keuangan di UKM Jurnalistik Paragraf Muda yang sebelumnya menggunakan Blogspot dan spreadsheet terpisah, sehingga terlihat bagian mana yang paling sering menimbulkan kendala.

Studi pustaka dilakukan dengan meninjau referensi terkait sistem informasi manajemen, organisasi kemahasiswaan, dan *penerapan Agile Scrum* dalam pengembangan perangkat lunak. Hasilnya digunakan sebagai dasar dalam merancang sistem dan menentukan fitur yang paling relevan dengan kebutuhan UKM jurnalistik.

## IV. HASIL DAN PEMBAHASAN

### 4.1 Gambaran Umum Implementasi Sistem

Pengembangan sistem Paragraf Muda dilakukan dengan pendekatan *Agile* menggunakan kerangka kerja *Scrum* karena pengembangan sistem untuk UKM jurnalistik cenderung dinamis: kebutuhan dapat berubah setelah pengurus melihat prototipe, dan banyak modul perlu diurutkan prioritasnya. Alih-alih merancang seluruh sistem sekaligus, pengembangan dibagi menjadi beberapa iterasi (*sprint*) yang masing-masing menghasilkan inkremen fitur yang dapat langsung diuji pengurus. Sebelum memasuki tahap Scrum, perancangan awal

p-ISSN : 2406-7733
e-ISSN : 2597-9922

sistem dilakukan menggunakan *Unified Modeling Language* (UML), lalu dilanjutkan dengan *Entity Relationship Diagram* (ERD) dan *Logical Record Structure* (LRS) untuk memastikan bahwa struktur data dan proses siap diimplementasikan pada tiap *sprint*. Kombinasi pendekatan ini memungkinkan pengembangan yang fleksibel namun tetap terstruktur.

#### 4.1.1 *Product Backlog*

Kebutuhan sistem dikumpulkan dan disusun ke dalam *Product Backlog* dengan prioritas penyelesaian pada modul utama redaksi, sebagaimana disajikan pada Tabel 1.

Tabel 1. *Product Backlog* Sistem Paragraf Muda

| ID *Backlog* | Kebutuhan Fitur | Prioritas |
|---|---|---|
| PB-01 | Manajemen hak akses (*role-based access control*) untuk memisahkan wewenang Litbang, Redaksi, Humas, dan Koordinator. | Tinggi |
| PB-02 | Fitur pembuatan *Brief* penugasan liputan/berita oleh divisi Litbang. | Tinggi |
| PB-03 | Fitur penulisan dan penyimpanan draf artikel oleh Anggota Redaksi berdasarkan *Brief* yang diberikan. | Tinggi |
| PB-04 | Fitur peninjauan (*review*) dan penyuntingan draf artikel oleh Kepala Redaksi. | Tinggi |
| PB-05 | Fitur persetujuan akhir (*approval*) dan publikasi artikel beserta *caption*-nya ke portal web oleh Koordinator. | Tinggi |
| PB-06 | Fitur pembuatan *Caption* dan konten penyerta oleh divisi Humas untuk naskah yang telah lolos tahap *review*. | Tinggi |

#### 4.1.2 Pelaksanaan *Sprint*

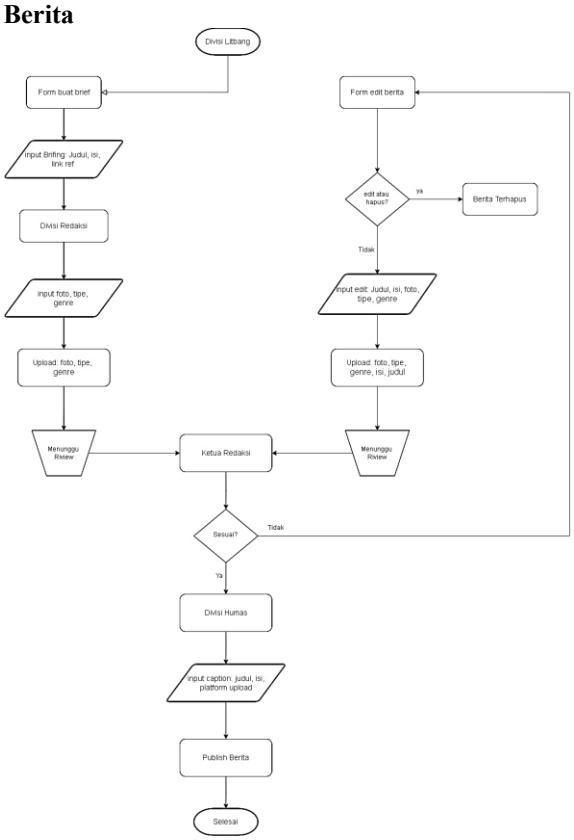
Penyelesaian *backlog* dilakukan melalui tahapan *Sprint* yang berfokus pada logika bisnis redaksional, seperti yang ditunjukkan pada Tabel 2.

Tabel 2. Pelaksanaan *Sprint* dan *Output*

| Iterasi | Fokus Penyelesaian | Hasil (*Output*) |
|---|---|---|
| *Sprint* 1 | PB-01 | Fondasi & Hak Akses: Menghasilkan modul autentikasi pengguna, manajemen hak akses (*role-based access control*), serta pembagian ruang kerja (*workspace*) dasar untuk divisi Litbang, Redaksi, Humas, dan Koordinator. |
| *Sprint* 2 | PB-02 dan PB-03 | Penugasan & Penulisan: Menghasilkan alur kerja hulu (*upstream*), di mana divisi Litbang dapat membuat *Brief* penugasan, dan Anggota Redaksi dapat menyetor draf naskah berita yang terhubung langsung dengan *brief* tersebut. |
| *Sprint* 3 | PB-04 dan PB-06 | Peninjauan & Pengayaan Konten: Menyelesaikan modul *review* bagi Kepala Redaksi untuk menyunting dan menyetujui naskah, dilanjutkan dengan modul bagi divisi Humas untuk menyusun *caption* media sosial berdasarkan naskah yang lolos *review*. |
| *Sprint* 4 | PB-05 | Persetujuan Akhir & Publikasi: Mengembangkan alur kerja hilir (*downstream*), berupa panel persetujuan (*approval*) final oleh Koordinator untuk menerbitkan naskah berita dan *caption* ke portal web utama. |

### 4.2 Perancangan Proses Fitur Berita

#### 4.2.1 *Flowchart* Proses Penulisan dan Publikasi Berita



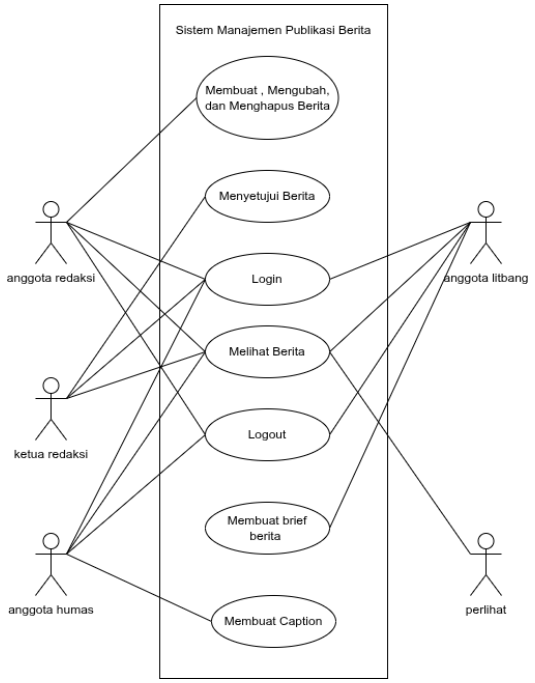

Gambar 1. *Flowchart* Penulisan dan Publikasi

p-ISSN : 2406-7733
e-ISSN : 2597-9922

Jika masih berstatus *draft*, berita hanya dapat dilihat dan diubah oleh penulisnya. Jika dikirim untuk *review*, berita akan masuk ke antrian koordinator redaksi Di sisi koordinator, *flowchart* menggambarkan alur pengambilan keputusan: koordinator membaca isi berita, kemudian menentukan apakah berita perlu direvisi atau sudah layak diterbitkan. Jika perlu revisi, berita dikembalikan kse penulis dengan status yang sesuai. Jika layak, koordinator menetapkan status berita menjadi published sehingga sistem menampilkan berita tersebut pada halaman portal Paragraf Muda. Dengan adanya *flowchart* ini, urutan proses dan titik keputusan dalam pengelolaan berita menjadi lebih mudah dipahami oleh tim pengembang maupun pengurus UKM.

### 4.2.2 *Use Case* Fitur Berita

Perancangan proses bisnis fitur berita diawali dengan pemodelan *use case* untuk menggambarkan aktor dan fungsi utama yang terlibat seperti yang diilustrasikan pada Gambar 2.




Gambar 2. *Use Case* Diagram Fitur Berita

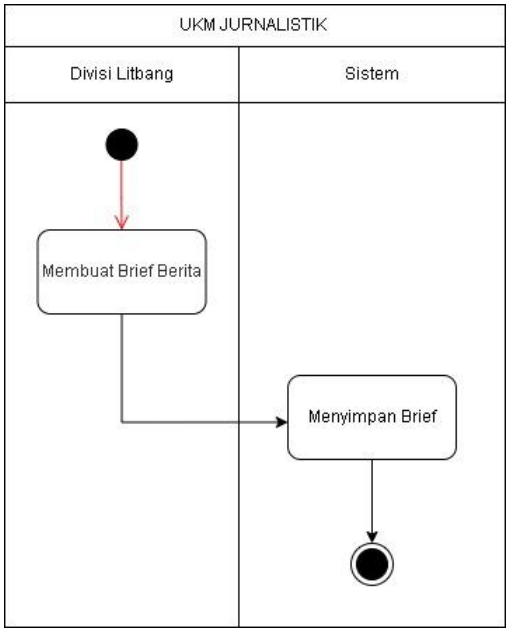
Pada *use case diagram* fitur berita, aktor yang terlibat antara lain anggota atau reporter, koordinator redaksi, dan admin sistem. Reporter memiliki hak untuk membuat berita baru, mengedit berita yang masih berupa *draft*, dan mengirimkan berita untuk di-*review*.

Koordinator redaksi memiliki hak untuk meninjau berita yang diajukan, menyetujui atau menolak berita, serta memublikasikan berita yang sudah layak tayang. Admin sistem bertugas mengelola akun pengguna dan, bila diperlukan, membantu pengelolaan berita yang mengalami kendala.

*Use case* ini menggambarkan beberapa fungsi utama, yaitu membuat berita, mengubah berita, mengirim berita untuk review, mereview berita, memublikasikan berita, serta mengarsipkan berita yang sudah tidak relevan. Dengan pemodelan ini, batasan peran dan hak akses setiap aktor menjadi lebih jelas, sehingga implementasi logika otorisasi di dalam aplikasi dapat mengikuti skenario yang digunakan di UKM jurnalistik.

### 4.2.3 *Activity* Diagram Penulisan dan Publikasi Berita

*Activity* diagram disusun untuk menggambarkan alur aktivitas dari proses penulisan hingga publikasi berita di portal Paragraf Muda. Untuk memperjelas alur kerja antar divisi, proses ini dijabarkan ke dalam beberapa tahapan aktivitas berdasarkan aktor yang terlibat. Proses kolaborasi ini diawali oleh Divisi Litbang. Pada tahap ini, anggota Litbang bertugas membuat *brief* berita yang berisi arahan atau topik liputan. *Brief* tersebut kemudian disimpan ke dalam sistem untuk diteruskan kepada tim redaksi, sebagaimana diperlihatkan pada Gambar 3.

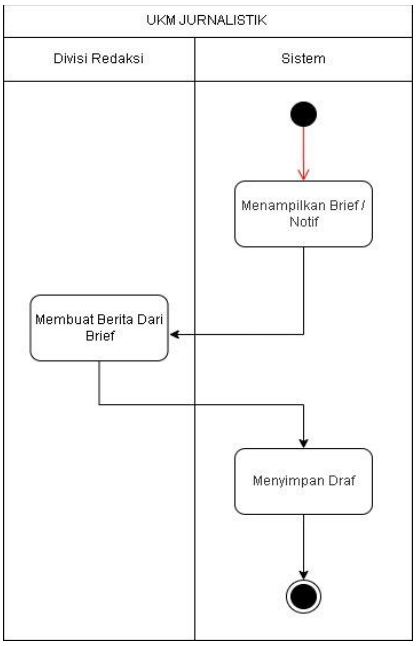



Gambar 3. *Activity* Diagram Divisi Litbang

Setelah *brief* tersimpan, sistem akan menampilkannya atau memberikan notifikasi kepada Divisi Redaksi. Proses dilanjutkan ketika anggota Divisi Redaksi melakukan login ke sistem, kemudian membuat berita berdasarkan *brief* tersebut. Penulis perlu mengisi formulir yang memuat judul berita, isi berita, kategori, dan melampirkan gambar bila diperlukan. Setelah formulir selesai diisi, Divisi Redaksi dapat menyimpan berita tersebut ke dalam sistem sebagai

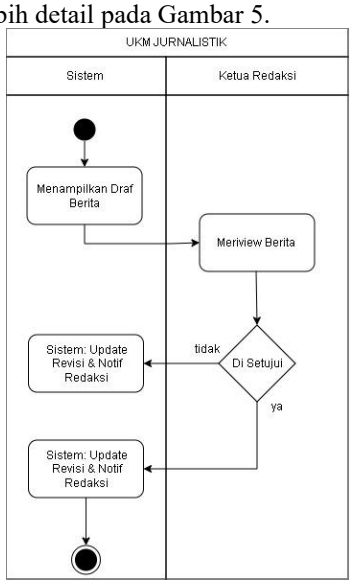
p-ISSN : 2406-7733
e-ISSN : 2597-9922

*draft* terlebih dahulu sebelum diajukan ke tahap selanjutnya, seperti yang diilustrasikan pada Gambar 4.



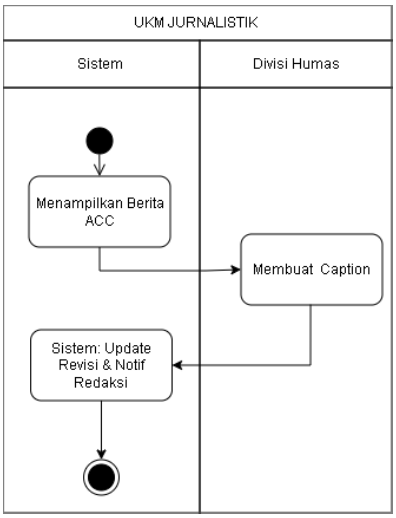

Gambar 4. *Activity* Diagram Divisi Redaksi

Berita yang telah berstatus *draft* kemudian akan masuk ke sistem dan ditampilkan pada daftar review milik Ketua Redaksi. Pada tahap persetujuan ini, Ketua Redaksi bertugas membaca isi berita, memeriksa kesesuaiannya dengan kaidah jurnalistik dan kebijakan UKM, lalu menentukan apakah berita tersebut disetujui atau tidak. Jika membutuhkan perbaikan, sistem akan melakukan pembaruan status revisi dan mengirimkan notifikasi kembali ke Divisi Redaksi. Sebaliknya, jika dinilai layak, Ketua Redaksi akan memberikan persetujuan (ACC) sehingga status berita diperbarui oleh sistem untuk diteruskan ke tahap akhir. Alur persetujuan ini dapat dilihat lebih detail pada Gambar 5.




Gambar 5. *Activity* Diagram Divisi Ketua Redaksi

Berita yang sudah mendapatkan persetujuan dari Ketua Redaksi selanjutnya akan ditampilkan oleh sistem kepada Divisi Humas. Pada tahapan ini, Divisi Humas bertugas membuat *caption* yang menarik untuk keperluan promosi berita. Setelah *caption* diselesaikan dan sistem diperbarui, sistem akan secara otomatis mempublikasikan dan menampilkan berita tersebut di halaman publik portal Paragraf Muda sebagaimana dijabarkan pada Gambar 6.




Gambar 6. *Activity* Diagram Divisi Humas

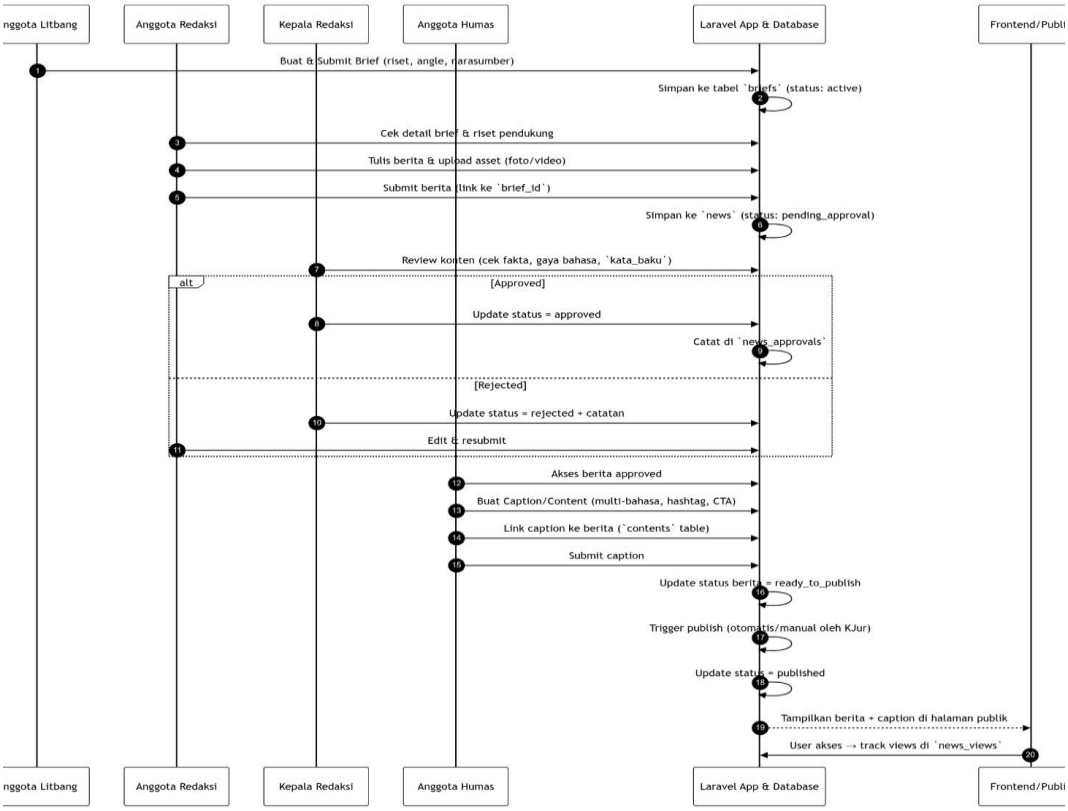
Secara keseluruhan, rangkaian *activity* diagram ini sangat krusial untuk membantu memastikan bahwa alur kerja yang berjalan di dalam sistem, mulai dari pra-penulisan hingga tahap publikasi, tetap terstruktur dan sesuai dengan prosedur operasional standar yang diharapkan oleh pengurus.

### 4.2.4 *Sequence Diagram* Proses Publish Berita

Sequence diagram seperti yang ditunjukkan pada Gambar 7 digunakan untuk memperlihatkan interaksi antara pengguna, antarmuka web, logika aplikasi, dan basis data saat berita dipublikasikan. Saat koordinator menekan tombol publish, antarmuka mengirim permintaan ke controller, controller memanggil fungsi pada model untuk mengubah status berita dan mencatat tanggal publikasi, kemudian basis data menyimpan perubahan tersebut.

Setelah itu, sistem mengirimkan respon kembali ke pengguna dalam bentuk halaman daftar berita yang sudah diperbarui. Dengan sequence diagram ini, urutan pesan dan objek yang terlibat dalam proses publish terdokumentasi dengan jelas sehingga mengurangi risiko kesalahan saat implementasi.

p-ISSN : 2406-7733
e-ISSN : 2597-9922



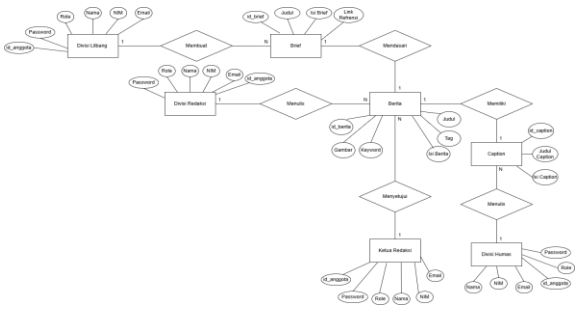

Gambar 7. Sequence Diagram Proses Publikasi Berita

### 4.3 Perancangan Basis Data: ERD dan LRS

#### 4.3.1 *Entity Relationship Diagram* (ERD)

Perancangan basis data sistem Paragraf Muda digambarkan melalui ERD seperti pada gambar 8 yang memuat entitas utama seperti *user*s, *news*, dan *categories*. Entitas *news* menyimpan informasi berita, meliputi judul, isi, tanggal pembuatan, tanggal publikasi, status, dan penulis. Entitas *users* menyimpan data pengguna dengan atribut nama, *email*, *password*, dan *role.*

Relasi satu-ke-banyak menghubungkan *user*s dengan *news*, karena satu pengguna dapat menulis beberapa berita, sedangkan setiap berita hanya memiliki satu penulis. Jika kategori digunakan, entitas categories juga berelasi satu-ke-banyak dengan *news*. Struktur ini memudahkan penelusuran berita berdasarkan penulis atau kategori, serta mendukung kebutuhan pelaporan UKM.

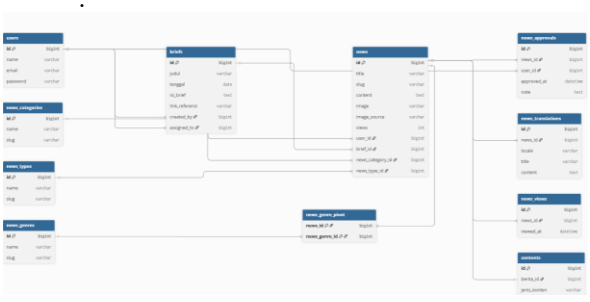


Gambar 8 *Entity Relationship Diagram* (ERD)

#### 4.3.2 *Logical Record Structure* (LRS)

LRS digunakan untuk menjabarkan desain ERD ke dalam struktur tabel yang akan diimplementasikan di MySQL. Tabel *news* dirancang dengan field utama seperti id_*news*, title, *slug*, *content*, *created*_at, *published*_at, *status*, *thumbnail*, *user*_id, dan *category*_id. Tipe data dipilih sesuai kebutuhan, misalnya text untuk isi berita dan datetime untuk tanggal publikasi.

Tabel *user*s memuat id_*user*, *name*, email, *password*, dan *role*. Hubungan antar tabel didefinisikan melalui foreign key, misalnya *user*_id pada tabel *news* yang mengacu ke id_*user* pada tabel *user*s. Dengan LRS yang jelas, penyusunan migrasi dan model pada Laravel menjadi lebih terarah dan meminimalkan perubahan struktur basis data selama proses pengembangan, dengan hasil pemetaan struktur data yang dapat dilihat pada Gambar 9



Gambar 9. *Logical Record Structure* (LRS)

p-ISSN : 2406-7733
e-ISSN : 2597-9922

### 4.4 Perancangan Antarmuka (*Wireframe*)

Tahap selanjutnya adalah merancang tampilan antarmuka melalui wireframe seperti yang diilustrasikan pada Gambar 10 Wireframe berfungsi sebagai kerangka visual untuk memastikan elemen dan navigasi sudah sesuai sebelum implementasi. Dan di rancangan ini fokus pada halaman Dashboard yang berfungsi sebagai pusat kendali bagi Administrator dan Koordinator Redaksi. Pada bagian atas area konten (A), terdapat empat kartu statistik yang menampilkan ringkasan data penting seperti total berita, *draft*, *pending review*, dan berita yang sudah tayang. Bagian tengah (B) menampilkan visualisasi aktivitas publikasi berita dengan fitur filter. Sementara bagian bawah (C) menyajikan dua grafik, yaitu grafik garis untuk melihat tren perkembangan dari bulan ke bulan dan grafik donat untuk menampilkan distribusi kategori berita. Dengan perancangan ini, diharapkan sistem Paragraf Muda dapat memberikan pengalaman pengguna yang intuitif dan informasi yang kompleks dapat disajikan secara visual.



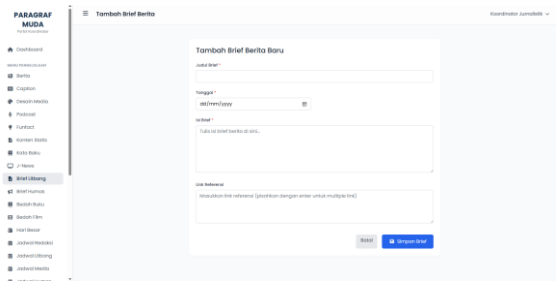
Gambar 10. *Wireframe Dashboard* Utama Sistem

### 4.5 Implementasi Fitur Berita Berbasis Web

#### 4.5.1 Halaman Input dan Manajemen *Brief*

Pada tahap implementasi, fitur manajemen instruksi kerja diwujudkan dalam bentuk halaman input yang memungkinkan Koordinator Jurnalistik menambah dan mengedit *brief* seperti yang ditunjukkan pada Gambar 11. Formulir yang disediakan mencakup judul *brief*, tanggal, isi *brief*, serta tautan atau link referensi pendukung peliputan. Setelah seluruh data diisi dan dikirim, sistem melakukan validasi sederhana untuk memastikan kolom wajib (bertanda bintang merah) telah terpenuhi sebelum menyimpan data ke dalam basis data.

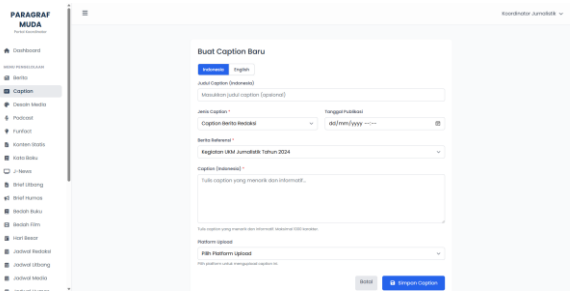
Koordinator Jurnalistik dapat mengelola daftar *brief* yang telah dibuat melalui menu manajemen ini, memantau distribusi instruksi kepada tim terkait, serta memperbarui informasi jika terdapat perubahan referensi liputan. Dengan adanya halaman manajemen *brief* ini, proses perencanaan awal redaksi, pemberian referensi, dan dokumentasi instruksi kerja di lingkungan Paragraf Muda menjadi lebih terstruktur dan efisien.



Gambar 11. Implementasi Antarmuka Halaman Tambah *Brief* Berita

#### 4.5.2 Halaman Input dan Manajemen Berita

Pada tahap implementasi, fitur berita diwujudkan dalam bentuk halaman input yang memungkinkan reporter menambah dan mengedit berita seperti yang ditunjukkan pada Gambar 12. Formulir yang disediakan mencakup judul, isi, kategori, unggah gambar, dan pengaturan status awal berita. Setelah data diisi dan dikirim, sistem melakukan validasi sederhana sebelum menyimpan data ke tabel *news*.



Gambar 12. Implementasi Antarmuka Halaman Tambah Berita

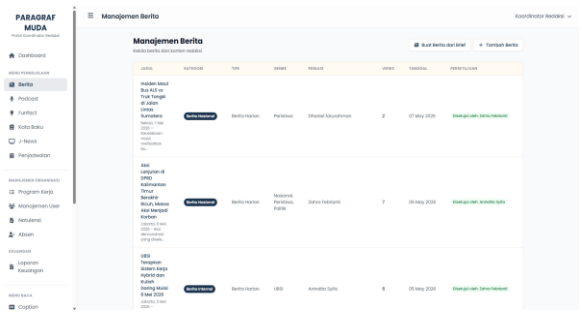
Reporter dapat melihat daftar berita yang pernah dibuat, memperbarui isi berita selama statusnya masih *draft*, dan mengirim berita tersebut ke koordinator untuk direview. Dengan adanya halaman manajemen berita ini, proses penulisan dan penyimpanan berita menjadi lebih terstruktur dan terdokumentasi.

#### 4.5.3 Proses Review dan Publikasi Berita

Koordinator redaksi memiliki halaman khusus untuk melihat daftar berita yang berstatus *draft* atau pending sebagaimana diperlihatkan pada Gambar 13. Dari halaman ini, koordinator dapat membuka detail berita, membaca isinya, dan memutuskan apakah berita perlu direvisi atau sudah siap dipublikasikan. Keputusan tersebut direalisasikan melalui tombol aksi yang mengubah status berita di sistem.Jika berita dinyatakan layak tayang, status diubah menjadi *published* dan sistem mencatat waktu publikasi. Berita yang sudah berstatus *published* kemudian berpindah dari daftar berita

p-ISSN : 2406-7733
e-ISSN : 2597-9922

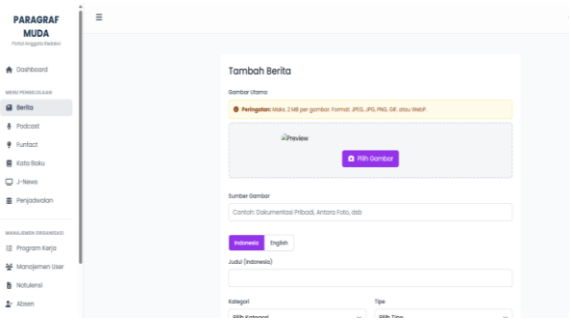
yang menunggu review ke daftar berita yang sudah terbit.




Gambar 13. Implementasi Antarmuka Halaman Manajemen Berita

### 4.5.4 Halaman Input dan Manajemen *Caption*

Pada tahap implementasi, fitur pengelolaan *caption* media sosial diwujudkan dalam bentuk halaman input yang memungkinkan Koordinator Jurnalistik membuat dan mengedit konten teks publikasi seperti yang ditunjukkan pada Gambar 14. Formulir ini dilengkapi dengan fitur alih bahasa (Indonesia dan English). Komponen input yang disediakan meliputi judul *caption* (opsional), jenis *caption*, tanggal publikasi, pemilihan berita referensi, isi teks *caption* (maksimal 1000 karakter), serta pilihan platform upload. Setelah data diisi, sistem akan melakukan validasi pada kolom wajib sebelum data disimpan melalui tombol "Simpan *Caption*".




Gambar 14. Implementasi Antarmuka Halaman Buat *Caption* Baru

Melalui halaman manajemen ini, Koordinator Jurnalistik dapat menyusun strategi penulisan teks publikasi yang selaras dengan berita referensi yang telah terdata di dalam sistem. Selain itu, integrasi penentuan platform dan tanggal publikasi membantu tim redaksi dalam menjadwalkan konten media sosial secara lebih teratur. Dengan adanya fitur ini, manajemen draf publikasi digital pada *platform* Paragraf Muda menjadi lebih terpusat, terstruktur, dan meminimalisasi risiko kesalahan redaksional sebelum diunggah ke media sosial resmi.

### 4.5.5 Tampilan Berita di Portal Paragraf Muda

Berita yang telah dipublikasikan ditampilkan di halaman utama portal Paragraf Muda seperti yang diilustrasikan pada Gambar 15. Sistem hanya menampilkan berita dengan status published dan mengurutkannya berdasarkan tanggal publikasi terbaru. Setiap item berita menampilkan judul, tanggal, penulis.




Gambar 15. Tampilan Detail Berita pada Portal Publik Paragraf Muda

Saat pengunjung memilih salah satu berita, sistem menampilkan halaman detail yang memuat isi berita lengkap serta informasi pendukung seperti kategori dan nama penulis. Dengan demikian, rangkaian proses mulai dari penulisan, *review*, hingga berita tayang di web menjadi satu alur yang utuh dan konsisten di dalam sistem.

### 4.6 Pengujian Sistem

Setelah tahap implementasi selesai, langkah selanjutnya adalah melakukan pengujian sistem untuk memastikan fungsionalitas perangkat lunak telah berjalan sesuai dengan spesifikasi kebutuhan yang dirancang. Pengujian dilakukan menggunakan metode pengujian penerimaan pengguna atau *User Acceptance Test* (UAT) dengan pendekatan *Black-Box Testing*. Metode ini dipilih karena berfokus pada evaluasi fungsionalitas sistem berdasarkan kesesuaian antara input dan output, tanpa perlu meninjau struktur kode internal aplikasi.

Pengujian ini melibatkan perwakilan pengurus dan anggota UKM Jurnalistik untuk memverifikasi skenario alur kerja utama, mulai dari proses autentikasi hak akses, pembuatan draf berita, hingga persetujuan akhir dan pub-likasi. Rincian skenario dan hasil pengujian sis-tem secara keseluruhan yang dilakukan oleh Alfian Eka Maulana pada 16 Februari 2026 dapat dilihat pada Tabel 3.

Tabel 3. Hasil Pengujian *Black-Box* Sistem UAT

| Tabel pengujian sistem secara keseluruhan | |
|---|---|
| Penguji: | Tanggal: |
| Alfian Eka Maulana | 16-02-2026 |
| | |

p-ISSN : 2406-7733
e-ISSN : 2597-9922

| No | Skenario & Input | Hasil & Status |
|---|---|---|
| 1 | Login Redaksi<br>Memastikan hak akses *role*.<br>Input: *Username* & *Password*. | Tampil *dashboard* sesuai *role*.<br><br>(Berhasil) |
| 2 | Buat *Brief* (Litbang)<br>Membuat penugasan liputan.<br>Input: Judul & Deskripsi. | *Brief* tersimpan & muncul di daftar tugas.<br><br>(Berhasil) |
| 3 | *Draft* Berita (Redaksi)<br>Pengisian naskah dari *brief*.<br>Input: Judul, Isi, Gambar. | Naskah tersimpan dgn status *Draft*.<br><br>(Berhasil) |
| 4 | Publikasi (Koordinator)<br>Persetujuan akhir naskah.<br>Input: Klik *'Publish'*. | Status menjadi *Published* & tayang.<br><br>(Berhasil) |
| 5 | *Caption* Sosmed (Humas)<br>Penambahan konten *caption*.<br>Input: Teks *caption*. | *Caption* tersimpan & terhubung.<br><br>(Berhasil) |

Berdasarkan hasil pengujian pada tabel di atas, seluruh skenario uji fungsionalitas menunjukkan status berhasil. Aplikasi mampu merespons masukan data dengan benar, membatasi hak akses sesuai dengan role pengguna, dan memperbarui status naskah berita secara akurat. Dengan demikian, dapat disimpulkan bahwa sistem informasi manajemen Paragraf Muda telah bebas dari kesalahan logika sistem (*bug*) dan dinyatakan layak untuk diimplementasikan guna mendukung operasional administrasi redaksional UKM.

## V. KESIMPULAN DAN SARAN

### 5.1 Kesimpulan

Setelah melalui seluruh tahapan perancangan, implementasi, serta pembahasan mengenai pengembangan platform Paragraf Muda, penelitian ini berhasil membuktikan bahwa sistem informasi berbasis web yang dibangun dengan *framework* Laravel 12 dan database MySQL mampu menjadi solusi terpusat bagi pengelolaan konten jurnalistik dan alur kerja redaksional UKM Jurnalistik. Sistem ini secara efektif menggantikan metode lama yang masih menggunakan *platform* terpisah seperti Blogspot dan spreadsheet, sehingga berhasil mengatasi kendala data yang tersebar dan sulit dilacak.

Selain itu, penggunaan metode Agile Scrum dengan pembagian kerja ke dalam beberapa iterasi *sprint* terbukti sangat adaptif dalam mengakomodasi kebutuhan organisasi yang dinamis, memastikan fitur-fitur utama mulai dari manajemen *brief* hingga publikasi dapat diimplementasikan secara bertahap sesuai prioritas. Melalui pembagian hak akses yang jelas antara divisi. Sistem ini telah mewujudkan alur kerja redaksional yang lebih profesional, transparan, dan terorganisir di lingkungan kampus.

### 5.2 Saran

Meskipun sistem telah berhasil berfungsi secara optimal, masih terdapat ruang untuk pengembangan lebih lanjut guna menyempurnakan performa *platform* di masa depan. Disarankan untuk menambahkan fitur notifikasi otomatis, seperti integrasi dengan *email gateway* atau WhatsApp API, agar proses koordinasi antara divisi dapat berjalan lebih cepat secara *real-time*. Di samping itu, mengingat tingginya mobilitas reporter di lapangan, pengembangan aplikasi ini ke dalam versi *mobile* atau optimasi sangat dianjurkan. Hal ni akan mempermudah *user* dalam mengakses sistem, melakukan peliputan, hingga mengirimkan *draft* berita secara langsung dari perangkat seluler tanpa terbatas oleh kendala perangkat *desktop*.

## DAFTAR PUSTAKA

Avianti, W., Muslimin, C. J., Kusumoningtyas, A. A., Muniroh, Junaedy, E., Tinia, A. G., Mulyati, C., Winasih, I., Suryaningsih, N., Safaria, S., Zairil, Jamhur, J. M., Alirejo, M. S., & Suhend, D. (2024). *Pengembangan Organisasi* (D. W. Mulyasari (Ed.)). Pradina Pustaka. Https://Www.Google.Co.Id/Books/Edition/Pengembangan_Organisasi/Kjureqaaqbaj?Hl=Jv&Gbpv=0&Kptab=Overview

Diyasa, I. G. S. M., Kraugusteeliana, & Noor, M. F. (2025). *Dasar Perencanaan Proyek Sains Data: Buku Ajar Berbasis Obe*. Thalibul Ilmi Publishing & Education. Https://Www.Google.Co.Id/Books/Edition/Dasar_Perencanaan_Proyek_Sains_Data_Buku/Ulz5eqaaqbaj?Hl=Jv&Gbpv=0

Hutauruk, A. C., & Pakpahan, A. F. (2021). Perancangan Sistem Informasi Organisasi Kemahasiswaan Berbasis Web Pada Universitas Advent Indonesia Menggunakan Metode Agile Development (Studi Kasus: Universitas Advent Indonesia). *Cogito Smart Journal* |, *7*(2), 2021.

Irfansyah, M. B., Harun, A., Pahlevi, M. N. R., Waheed, B., Fariza, A., & Rante, H. (2026). *Implementasi Metodologi Agile Scrum Dalam Pengembangan Aplikasi Vsehat*. *14*(2), 11. Https://Journal.Eng.Unila.Ac.Id/Index.Php/Jitet/Article/View/9251

Magdalena, L. (2023). *Scrum Agile: Optimalisasi Kualitas Produk Manajemen* (E. Efitra & N. Safitri (Eds.)). Pt. Sonpedia Publishing Indonesia. Https://Www.Google.Co.Id/Books/Edition/Scrum_Agile_Optimalisasi_Kualitas_Produk/5exme

p-ISSN : 2406-7733
e-ISSN : 2597-9922

aaaqbaj?Hl=Jv&Gbpv=0

Namruddin, R., Basalamah, A., Ali, M. Z. A., Syarifuddin, A., Alam, S., Wardhani, N., & Abdurrahman, T. S. D. (2023). *Belajar Database Dengan Mudah Menggunakan Mysql* (R. Namruddin & A. Basalamah (Eds.)). Cv. Tohar Media. Https://Www.Google.Co.Id/Books/Edition/Belajar_Database_Dengan_Mudah_Menggunaka/Mhneeaaaqbaj?Hl=Jv&Gbpv=0

Nasution, N. T. S., Astria, R., Ramadhan, S., Ramadhani, W., & Dhany, H. W. (2024). Penerapan Absensi Digital Mengenai Kehadiran Pengunjung Pojok Literasi Berbasis Web Di Brida Kota Medan. *Jurnal Minfo Polgan*, *13*(2), 2090–2095. Https://Doi.Org/10.33395/Jmp.V13i2.14389

Patma, T. S., Maskan, M., & Utaminingsih, A. (2018). *Sistem Informasi Manajemen*. Polinema Press. Https://Www.Google.Co.Id/Books/Edition/Sistem_Informasi_Manajemen/Asjydwaaqbaj?Hl=Jv&Gbpv=0

Romadloni, N. T., & Miswanto, M. (2024). *Sistem Basis Data : Teori Dan Praktikum* (E. Efitra & E. Pamela (Eds.)). Pt. Sonpedia Publishing Indonesia. Https://Www.Google.Co.Id/Books/Edition/Sistem_Basis_Data_Teori_Dan_Praktikum/Roekeqaaqbaj?Hl=Jv&Gbpv=0

Sihombing, R. A., Sulistyohati, A., Izzatillah, M., & Natsir, F. (2025). *Kuasai Uml Dalam 30 Menit : Rahasia Cepat Membuat Sistem Pemilihan Karyawan Terbaik* (R. Harun (Ed.)). Pt Penerbit Qriset Indonesia. Https://Www.Google.Co.Id/Books/Edition/Kuasai_Uml_Dalam_30_Menit_Rahasia_Cepat/-W-Veqaaqbaj?Hl=Jv&Gbpv=0

Sophan, M. K., Yusuf, M., & Satoto, B. (2025). *Level Up Coding: Pemrograman Aplikasi Web Dengan Php Dan Mysql*. Media Nusa Creative (Mnc Publishing). Https://Www.Google.Co.Id/Books/Edition/Level_Up_Coding_Pemrograman_Aplikasi_Web/Kfkieqaaqbaj?Hl=Jv&Gbpv=0

Suprapto, U. (2021). *Pemodelan Perangkat Lunak Smk/Mak Kelas Xi*. Gramedia Widiasarana Indonesia. Https://Www.Google.Co.Id/Books/Edition/Pemodelan_Perangkat_Lunak_Smk_Mak_Kelas/Kn8teaaaqbaj?Hl=Jv&Gbpv=0

Trisanto, D., Rismawati, N., Izzatillah, M., & Femy Mulya, M. (2023). Analisis Dan Perancangan Sistem Informasi E-Learning Menggunakan Metode Scrum Berbasis Framework Laravel Dan Bootstrap. *Journal Of Information System, Applied, Management, Accounting And Research*, *7*(2), 225–232. Https://Doi.Org/10.52362/Jisamar.V7i2.1052

Winarti, W., & Albaqir, A. H. (2025). *Aplikasi Manajemen Penjualan Dan Inventori Pada Toko Ritel (Teori, Perancangan Dan Implementasi Sistem Berbasis Waterfall)*. Cv Eureka Media Aksara. Https://Www.Google.Co.Id/Books/Edition/Aplikasi_Manajemen_Penjualan_Dan_Invento/Q4oeeqaaqbaj?Hl=Jv&Gbpv=0

Yudhanto, Y., & Prasetyo, H. A. (2019). *Mudah Menguasai Framework Laravel*. Elex Media Komputindo. Https://Www.Google.Co.Id/Books/Edition/Mudah_Menguasai_Framework_Laravel/8tkddwaaqbaj?Hl=Jv&Gbpv=0&Kptab=Overview