

RANCANG BANGUN WEBSITE PENGELOLAAN BUKU DIGITAL BERBASIS *NATURAL LANGUAGE PROCESSING* (NLP)

Rangga Wahyu Pratama^{1*}, Paskah Abadi Simanullang², Peter Tymoty Hutabarat³,
Revidamurti Dly⁴, Adidtya Perdana⁵

Program Studi Ilmu Komputer, Universitas Negeri Medan, Sumatera Utara, Indonesia¹

Email* : Ranggawahyupratama386@mhs.unimed.ac.id

Program Studi Ilmu Komputer, Universitas Negeri Medan, Sumatera Utara, Indonesia²

Email: paskahsimanullang24@gmail.com

Program Studi Ilmu Komputer, Universitas Negeri Medan, Sumatera Utara, Indonesia³

Email: pertertymotyhutabarat08@gmail.com

Program Studi Ilmu Komputer, Universitas Negeri Medan, Sumatera Utara, Indonesia⁴

Email: revidamurti@gmail.com

Program Studi Ilmu Komputer, Universitas Negeri Medan, Sumatera Utara, Indonesia⁵

Email: adidtya@unimed.ac.id

(*) *Corresponding Author*

ABSTRAK

Sistem pengelolaan buku digital menjadi solusi penting di era digital untuk membantu pengguna mengorganisasi dan mencari koleksi buku dengan efisien. Dalam konteks perkembangan teknologi informasi, kebutuhan akan *platform* terintegrasi yang dapat menyimpan data buku serta memberikan kemudahan akses melalui fitur pencarian berbasis metadata semakin meningkat. Namun, permasalahan utama yang dihadapi adalah belum adanya sistem yang mendukung ekstraksi informasi otomatis dari file buku digital dan pencarian berdasarkan kriteria seperti judul, bahasa, atau topik. Untuk mengatasi hal tersebut, penelitian ini menggunakan metode *Research and Development* (R&D) untuk merancang sistem, studi literatur untuk menganalisis teknologi NLP, serta observasi untuk memahami kebutuhan pengguna. Hasil penelitian menghasilkan aplikasi berbasis PHP, MySQL, dan Python (NLP) dengan fitur unggah *file*, ekstraksi metadata otomatis, pencarian berdasarkan kriteria tertentu, serta manajemen *profil* pengguna. Kesimpulan dari penelitian ini menunjukkan bahwa sistem yang dikembangkan mampu memberikan solusi praktis dan efisien dalam pengelolaan buku digital, dengan desain modular yang mendukung skalabilitas dan pengembangan lebih lanjut.

Kata Kunci: Buku, NLP, *website*

ABSTRACT

A digital book management system has become a crucial solution in the digital era, helping users efficiently organize and search their book collections. In the context of information technology advancements, the demand for an integrated platform that can store book data and provide easy access through metadata-based search features is increasing. However, the main challenge lies in the absence of a system that supports automatic information extraction from digital book files and searches based on criteria such as title, language, or topic. To address this issue, this study employs the Research and Development (R&D) method to design the system, a literature review to analyze NLP technology, and observations to understand user needs. The research results in an

application built with PHP, MySQL, and Python (NLP), featuring file uploads, automatic metadata extraction, search based on specific criteria, and user profile management. The study concludes that the developed system provides a practical and efficient solution for digital book management, with a modular design that supports scalability and further development.

Keywords: Book, NLP, website

1. PENDAHULUAN

Dalam era digital, pengelolaan buku tidak hanya sekedar menyimpan dan mengorganisir dokumen, tetapi juga memastikan kemudahan akses serta pencarian informasi yang lebih efektif, penggunaan teknologi informasi dalam perpustakaan modern memungkinkan sistem menjadi lebih responsif terhadap kebutuhan pengguna dengan memanfaatkan kecerdasan buatan, termasuk *Natural Language Processing (NLP)*[1]. Ini menunjukkan penerapan teknologi NLP dapat menjadi solusi dalam meningkatkan efisiensi pengelolaan buku digital. Implementasi sistem informasi berbasis *web* berperan penting dalam meningkatkan efektivitas pengelolaan dokumen, memungkinkan pengguna untuk mengakses data secara fleksibel kapan pun dan di mana pun, serta meminimalkan potensi kehilangan arsip yang kerap terjadi akibat metode manual yang masih diterapkan di banyak instansi. Berdasarkan hal tersebut, penelitian ini berfokus pada pengembangan website untuk pengelolaan buku digital dengan dukungan *Natural Language Processing (NLP)*.

Website pengelolaan buku digital dirancang untuk menyederhanakan proses pencarian, mengelompokkan buku secara otomatis, serta memberikan rekomendasi yang lebih relevan, sehingga pengguna dapat lebih mudah menavigasi dan mengatur koleksi buku digital [2]. Klasifikasi buku berbasis AI mampu mengidentifikasi pola dalam *teks* secara otomatis, sehingga pengelompokan buku dapat dilakukan dengan lebih cepat dan akurat[3]. Sistem berbasis NLP dapat mengurutkan berdasarkan abjad dengan tingkat akurasi yang lebih tinggi dibandingkan metode manual. *Natural Language Processing (NLP)* adalah cabang dari kecerdasan buatan yang berfokus pada pemrosesan teks dan bahasa alami manusia. Dalam konteks perpustakaan digital, NLP berperan dalam meningkatkan efektivitas pencarian dan klasifikasi dokumen.

Penerapan NLP dalam perpustakaan modern mampu mengatasi tantangan peningkatan volume informasi digital, meningkatkan efisiensi layanan, serta memperbaiki interaksi pengguna dengan koleksi buku[4]. Contoh implementasi NLP yaitu dalam sistem pencarian berbasis semantik. Peminjaman buku secara digital melalui aplikasi berbasis NLP memungkinkan pengguna untuk menemukan koleksi yang dibutuhkan dengan lebih cepat dan relevan terutama ketika koleksi fisik terbatas[5]. NLP dapat membantu dalam pengelolaan buku digital dengan meningkatkan pengalaman pengguna dalam mengakses informasi. Dalam penerapan NLP, terdapat beberapa algoritma yang umum digunakan untuk klasifikasi dan pengelompokan buku. Penggunaan metode *Naive Bayes Classifier* dalam klasifikasi genre buku mampu meningkatkan akurasi sistem dalam menentukan kategori buku berdasarkan *teks* sinopsisnya[3].

Metode *Term-Frequency Inverse Document Frequency (TF-IDF)* juga sering digunakan untuk menilai tingkat kepentingan kata dalam suatu dokumen. Selain algoritma statistik, teknik *deep learning* seperti *Artificial Neural Network (ANN)* juga mulai diterapkan dalam klasifikasi buku digital. Penerapan ANN dalam NLP mampu mengenali pola bahasa yang lebih kompleks, sehingga sistem dapat memberikan hasil

klasifikasi yang lebih akurat dibandingkan metode konvensional[6]. Pemilihan algoritma yang tepat sangat bergantung pada kompleksitas sistem dan tujuan akhir dari pengelolaan buku digital. Penggunaan NLP dalam sistem pengelolaan buku digital, diperlukan berbagai teknologi pendukung. Integrasi basis data yang kuat seperti MongoDB dan PostgreSQL sangat diperlukan untuk menyimpan metadata buku serta hasil analisis NLP. Selain itu, Penggunaan *cloud storage* seperti AWS S3 dan *Google Drive* API juga menjadi pilihan utama dalam penyimpanan skala besar[7].

Beberapa *framework* yang sering digunakan dalam pengembangan NLP antara lain adalah *TensorFlow*, *PyTorch*, *NLTK*, dan *spaCy*. Pemilihan *framework* harus disesuaikan dengan kebutuhan sistem, mengingat masing-masing memiliki keunggulan tersendiri dalam menangani pemrosesan bahasa alami[8]. Pengelolaan buku digital berbasis NLP memerlukan kombinasi berbagai teknologi untuk memastikan kinerja sistem yang optimal. Keamanan data dalam sistem pengelolaan buku digital juga menjadi faktor penting yang perlu diperhatikan. Sistem enkripsi berbasis AES dan *hashing password* dapat diterapkan untuk melindungi akses pengguna terhadap dokumen yang bersifat pribadi atau terbatas[9]. Penerapan metode autentikasi berbasis *Oauth* juga disarankan untuk memastikan bahwa hanya pengguna yang memiliki izin yang dapat mengakses koleksi tertentu. Penelitian yang dilakukan oleh Mubarak dan Abdi, menunjukkan bahwa penerapan indeks pencarian berbasis *NLP* mampu mengurangi waktu pencarian buku digital hingga 70% dibandingkan metode pencarian tradisional berbasis kata kunci. Sistem berbasis NLP tidak hanya meningkatkan keamanan data, tetapi juga mempercepat pencarian dan akses terhadap koleksi buku digital[10]

2. METODE

Teknik Pengumpulan Data

Data yang digunakan dalam penelitian ini diperoleh melalui:

1. Studi Literatur.
Mengumpulkan informasi dari jurnal, buku, dan artikel terkait NLP serta sistem pengelolaan buku digital.
2. Observasi
Mengamati sistem yang telah ada untuk memahami permasalahan yang dihadapi pengguna.

Metode Pengembangan Sistem

Pengembangan sistem dalam penelitian ini dilakukan dengan tahapan sebagai berikut:

1. Analisis Kebutuhan
 - 1) Mengidentifikasi kebutuhan pengguna terkait pengelolaan buku digital, seperti fitur *upload*, pencarian, pengelompokan (berdasarkan bahasa, topik, atau huruf awal judul), serta aksi lihat, *download*, dan hapus
 - 2) Menganalisis kebutuhan sistem dari segi perangkat lunak (PHP, Python, MySQL) dan perangkat keras (server lokal menggunakan XAMPP).
2. Perancangan Database
 - 1) Merancang *database* MySQL untuk menyimpan data pengguna dan metadata buku, termasuk informasi seperti ID unik, judul, bahasa, topik, lokasi *file*, dan waktu unggah.
 - 2) Mendesain antarmuka pengguna (UI/UX) yang ramah pengguna. Menggunakan struktur folder lokal (*uploads*) dengan organisasi berbasis ID

unik untuk menyimpan *file* PDF, gambar hasil konversi, *teks* hasil OCR, dan metadata *JSON*.

3. Perancangan Arsitektur

- 1) Menggunakan arsitektur 3-tier yang terdiri dari:
 - a. *Frontend*: HTML, CSS, dan JavaScript untuk antarmuka pengguna.
 - b. *Backend*: PHP untuk logika aplikasi dan interaksi dengan *database*.
 - c. *Database*: MySQL untuk penyimpanan data pengguna dan metadata buku.
- 2) Mendesain antarmuka pengguna (UI/UX) yang ramah pengguna. Memastikan aspek keamanan dengan autentikasi pengguna menggunakan *session* PHP, efisiensi dengan pengelompokan *file* berdasarkan ID unik, dan skalabilitas untuk pengembangan di masa mendatang.

4. Perancangan Alur Web

- 1) Mengembangkan sistem berbasis *web* dengan PHP sebagai *backend* dan *Python* untuk otomatisasi proses seperti konversi file (DOCX ke PDF), OCR, dan ekstraksi metadata menggunakan NLP.
- 2) Menyediakan penyimpanan lokal sementara untuk *file* pengguna dengan kemungkinan integrasi *cloud storage* di masa mendatang guna meningkatkan kapasitas penyimpanan

3. HASIL DAN PEMBAHASAN

Analisis Kebutuhan

Pengembangan sistem pengelolaan buku digital ini dilatarbelakangi oleh kebutuhan untuk menyediakan solusi yang efisien dalam mengorganisir, mencari, dan mengakses koleksi buku secara digital. Sistem ini dirancang untuk memenuhi kebutuhan pengguna yang memerlukan *platform* terintegrasi untuk mengunggah, mengelompokkan, dan mengelola file buku berdasarkan kategori seperti bahasa, topik, atau huruf awal judul. Selain itu, kebutuhan akan fitur pencarian yang cepat dan akurat menjadi salah satu fokus utama agar pengguna dapat menemukan buku dengan mudah berdasarkan kata kunci tertentu. Dalam konteks penggunaan *web*, antarmuka yang responsif, intuitif, dan ramah pengguna menjadi prioritas untuk memastikan bahwa setiap interaksi dengan sistem dapat dilakukan tanpa hambatan.

Dalam proses pengembangan sistem, kebutuhan teknis juga dianalisis secara mendalam. Dari segi perangkat lunak, diperlukan integrasi teknologi seperti PHP untuk *backend*, Python untuk otomatisasi proses (konversi *file*, OCR, dan ekstraksi metadata), serta MySQL untuk penyimpanan data terstruktur. Dari segi perangkat keras, sistem diimplementasikan pada *server* lokal menggunakan XAMPP untuk memastikan ketersediaan sumber daya yang memadai selama tahap pengembangan. Selain itu, kebutuhan akan keamanan data, efisiensi pemrosesan, dan skalabilitas juga menjadi pertimbangan penting agar sistem dapat diandalkan dan mampu berkembang sesuai dengan peningkatan jumlah pengguna dan data di masa mendatang. Dengan analisis kebutuhan ini, sistem dirancang untuk memberikan solusi komprehensif yang tidak hanya memenuhi kebutuhan pengguna saat ini tetapi juga memiliki potensi untuk dikembangkan lebih lanjut.

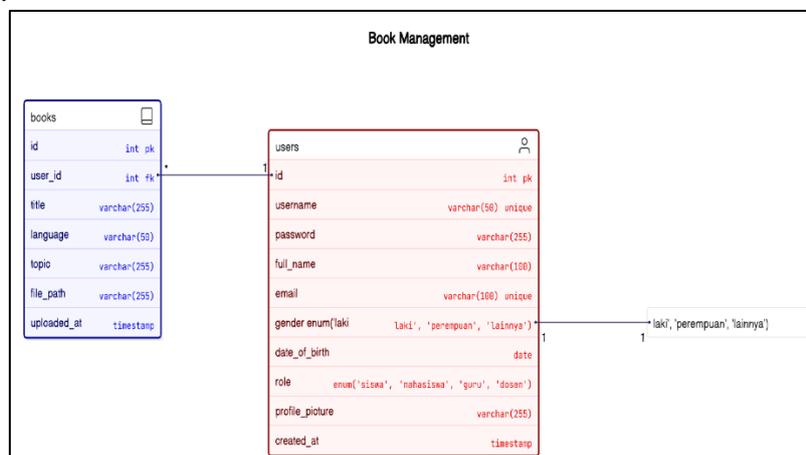
Perancangan Database

Database yang digunakan dalam sistem ini diberi nama *book_management* dan dibangun menggunakan aplikasi XAMPP dengan *server* MySQL. *Database* ini dirancang untuk mendukung fungsionalitas utama sistem, seperti pengelolaan data pengguna dan

informasi buku digital secara terstruktur. Dalam *database book_management*, terdapat dua tabel utama, yaitu *users* dan *books*, yang masing-masing memiliki peran penting dalam menjaga integritas dan efisiensi data. Tabel *users* digunakan untuk menyimpan informasi pengguna yang terdaftar dalam sistem. Tabel ini mencakup beberapa kolom penting, seperti *id* sebagai *primary key* untuk mengidentifikasi setiap pengguna secara unik, serta *username* dan *password* yang digunakan untuk proses autentikasi *login*. Selain itu, tabel ini juga mencakup kolom tambahan seperti *full_name*, *email*, *gender*, *date_of_birth*, dan *role*, yang memberikan detail profil pengguna secara komprehensif.

Kolom *profile_picture* digunakan untuk menyimpan lokasi *file* gambar profil pengguna, sementara kolom *created_at* mencatat waktu pendaftaran akun. Struktur ini memastikan bahwa data pengguna dapat dikelola dengan aman dan mudah diakses saat diperlukan. Sementara itu, tabel *books* dirancang untuk menampung informasi terkait buku digital yang diunggah oleh pengguna. Tabel ini mencakup kolom *id* sebagai *primary key* untuk mengidentifikasi setiap entri buku secara unik, serta *user_id* sebagai *foreign key* yang menghubungkan setiap buku dengan pengguna yang mengunggahnya. Kolom lainnya mencakup *title* untuk menyimpan judul buku, *language* untuk bahasa buku, *topic* untuk topik atau kategori buku, *file_path* untuk menyimpan lokasi *file* buku (misalnya PDF), dan *uploaded_at* untuk mencatat waktu pengunggahan.

Struktur tabel ini memungkinkan pengelompokan dan pencarian buku berdasarkan kriteria tertentu, seperti huruf awal judul, bahasa, atau topik, sehingga memudahkan pengguna dalam mengakses koleksi buku. Dengan desain *database* yang terstruktur ini, sistem mampu mendukung berbagai fitur utama, seperti pengelolaan data pengguna secara aman dan efektif, serta pencarian dan pengelompokan buku berdasarkan kriteria tertentu. Desain ini juga memastikan bahwa data dapat disimpan, diambil, dan dikelola dengan efisien, sehingga meningkatkan performa dan keandalan sistem secara keseluruhan.



Gambar 1. ERD sistem

Arsitektur Sistem

Sistem *Book Management System* dirancang dengan arsitektur modular yang menggabungkan berbagai teknologi seperti HTML, CSS, JavaScript, PHP, dan Python untuk menciptakan pengalaman pengguna yang interaktif, efisien, dan terorganisir. HTML dan CSS digunakan sebagai dasar untuk membangun antarmuka pengguna yang responsif dan estetis. HTML menyediakan struktur halaman dengan elemen-elemen seperti formulir *input*, tabel data, tombol aksi, dan notifikasi, sementara CSS memberikan *styling* pada elemen-elemen tersebut untuk memastikan tampilan yang konsisten dan

menarik. Misalnya, tombol aksi seperti lihat, *download* dan hapus diberi warna dan efek *hover* yang sesuai untuk meningkatkan kegunaan dan estetika, serta memastikan bahwa desain tetap responsif di berbagai perangkat. JavaScript memainkan peran penting dalam menambahkan interaktivitas dan fungsionalitas dinamis pada halaman web. Pada halaman Login.php, JavaScript digunakan untuk mengimplementasikan fitur *toggle password visibility* melalui fungsi `togglePassword()`, yang memungkinkan pengguna untuk menampilkan atau menyembunyikan kata sandi dengan mengubah atribut *type* dari elemen *input password* dan mengubah *teks* tombol menjadi "Hide" atau "Show".

Pada halaman Profile.php, JavaScript digunakan untuk mengelola mode edit profil dengan menambahkan *event listener* pada tombol "Edit Profil" dan "Batal". Ketika tombol "Edit Profil" ditekan, informasi profil statis disembunyikan dan formulir edit ditampilkan, sementara tombol "Batal" mengembalikan tampilan ke informasi profil awal. Fitur ini memungkinkan pengguna untuk dengan mudah mengedit data pribadi pengguna tanpa harus memuat ulang halaman. Di halaman Signup.php, JavaScript mendukung pemilihan *file* dari perangkat atau kamera, pratinjau gambar, dan notifikasi dinamis menggunakan WebRTC API untuk akses kamera dan *FileReader* API untuk konversi gambar ke format base64.

PHP bertindak sebagai backend untuk memproses data yang dikirim dari *frontend* dan berinteraksi dengan *database*. Halaman-halaman seperti Index.php, Signup.php, Login.php, dan Dashboard.php berisi logika untuk mengelola sesi pengguna, autentikasi, dan operasi CRUD (*Create, Read, Update, Delete*) pada data buku. PHP memvalidasi input pengguna saat *login* atau *signup*, menyimpan data pengguna ke *database*, dan mengelola unggahan *file* PDF. Pada halaman By Language.php dan By Topic.php, PHP mengambil data dari *database* dan mengelompokkannya berdasarkan bahasa atau topik tertentu untuk ditampilkan dalam tabel yang terorganisasi. Selain itu, PHP juga memanggil skrip Python untuk melakukan tugas-tugas otomatisasi menggunakan perintah `exec()`.

Python digunakan untuk berbagai tugas otomatisasi dan pemrosesan data, yang mendukung fungsi-fungsi inti sistem. Skrip Python seperti `docx2pdf` digunakan untuk mengonversi file DOCX menjadi PDF menggunakan fungsi `convert`. Skrip `pdf2image` mengonversi halaman pertama PDF menjadi gambar menggunakan fungsi `convert_from_path`. Untuk ekstraksi teks dari gambar, sistem menggunakan `surya.recognition` dan `surya.detection` yang merupakan bagian dari library OCR untuk mengenali *teks* dari gambar hasil konversi. Selain itu, sistem juga menggunakan model AI dari *OpenAI* untuk mengekstrak metadata seperti judul, bahasa, dan topik dari teks yang diekstrak. Metadata ini kemudian disimpan dalam format JSON untuk digunakan dalam pengelompokan dan pencarian data buku.

Direktori `assets/` berisi *file-file* statis seperti CSS, JavaScript, dan gambar yang mendukung tampilan dan fungsionalitas aplikasi. Direktori `uploads/` digunakan untuk menyimpan *file-file* yang diunggah oleh pengguna, seperti *file* PDF, gambar hasil konversi, dan *file teks* hasil OCR. *File-file* ini dikelompokkan berdasarkan ID pengguna untuk memastikan privasi dan organisasi data. Dengan kombinasi teknologi ini, sistem mampu menyediakan pengalaman pengguna yang interaktif, aman, dan efisien, mulai dari proses pendaftaran hingga pengelolaan buku berdasarkan kategori seperti bahasa dan topik. Arsitektur sistem ini dirancang dengan pendekatan modular, di mana setiap komponen memiliki peran spesifik untuk memastikan fungsi dan tampilan sistem bekerja secara harmonis, sehingga pengguna dapat dengan mudah mengelola dan mengakses data buku sesuai kebutuhan.

```
C:\MANAJEMEN BUKU NLP
| book_management.sql # File SQL untuk membuat dan mengelola database sistem.
| by_language.php # Halaman PHP untuk menampilkan daftar buku berdasarkan bahasa.
| by_titles_initial.php # Halaman PHP untuk menampilkan daftar buku berdasarkan huruf awal judul.
| by_topic.php # Halaman PHP untuk menampilkan daftar buku berdasarkan topik.
| dashboard.php # Halaman utama (dashboard) untuk mengelola buku, termasuk unggah dan penghapusan.
| database.php # File PHP yang berisi konfigurasi koneksi ke database.
| index.php # Halaman utama aplikasi, berfungsi sebagai landing page dan login.
| login.php # Halaman untuk autentikasi pengguna (login).
| logout.php # Script PHP untuk keluar dari sesi pengguna.
| profile.php # Halaman untuk mengelola profil pengguna, seperti edit informasi akun.
| signup.php # Halaman untuk pendaftaran akun baru.
|
| ---assets # Direktori untuk menyimpan file statis seperti CSS dan JavaScript.
| | ---css # Subdirektori untuk file CSS.
| | | by_language.css # File CSS untuk halaman by_language.php.
| | | by_titles_initial.css # File CSS untuk halaman by_titles_initial.php.
| | | by_topic.css # File CSS untuk halaman by_topic.php.
| | | dashboard.css # File CSS untuk halaman dashboard.php.
| | | index.css # File CSS untuk halaman index.php.
| | | login.css # File CSS untuk halaman login.php.
| | | profile.css # File CSS untuk halaman profile.php.
| | | signup.css # File CSS untuk halaman signup.php.
| |
| | ---python_scripts # Direktori untuk skrip Python yang digunakan dalam otomatisasi.
| | | docx_to_pdf.py # Skrip Python untuk mengonversi file DOCX menjadi PDF.
| | | get_metadata.py # Skrip Python untuk mengekstrak metadata (judul, bahasa, topik) dari teks menggunakan model AI.
| | | image_to_text.py # Skrip Python untuk mengekstrak teks dari gambar menggunakan OCR.
| | | pdf_to_image.py # Skrip Python untuk mengonversi halaman pertama PDF menjadi gambar.
| |
| | ---uploads # Direktori untuk menyimpan file yang diunggah oleh pengguna.
| | | ---books # Subdirektori untuk file buku (PDF) yang diunggah.
| | | | ---user_18 # Direktori khusus untuk pengguna dengan ID 18.
| | | | | 67f5062f2e9c0.pdf # Contoh file PDF yang diunggah oleh pengguna.
| | | |
| | | | ---metadata # Subdirektori untuk file metadata dalam format JSON.
| | | | | ---user_18 # Direktori khusus untuk metadata pengguna dengan ID 18.
| | | | | | 67f5062f2e9c0.json # Contoh file JSON yang berisi metadata buku.
| | | | |
| | | | ---ocr # Subdirektori untuk file hasil pemrosesan OCR.
| | | | | ---books_picture # Subdirektori untuk gambar hasil konversi PDF.
| | | | | | ---user_18 # Direktori khusus untuk gambar pengguna dengan ID 18.
| | | | | | | 67f5062f2e9c0.jpg # Gambar halaman pertama PDF.
| | | | | | | page1.jpg # Contoh gambar hasil konversi lainnya.
| | | | | |
| | | | | ---text # Subdirektori untuk file teks hasil ekstraksi OCR.
| | | | | | ---user_18 # Direktori khusus untuk file teks pengguna dengan ID 18.
| | | | | | | 67f5062f2e9c0.txt # Contoh file teks hasil ekstraksi OCR.
| | | | |
| | | | ---pictures # Subdirektori untuk gambar umum, seperti foto profil pengguna.
| | | | | Leonardo_Vision_XL_Blue_Ducaty_motorbike_parked_in_the_city_v1_2.jpg # Contoh gambar umum.
| | | | | premium_photo-1689568126014-06fea9d5d341.jpeg # Contoh gambar lainnya.
| |
|
|
```

Gambar 2. Struktur folder sistem

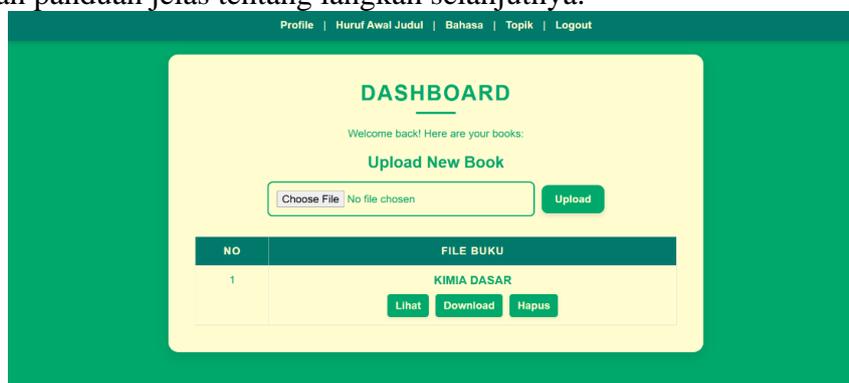
Alur Web

Halaman pertama sistem, 'index.php' adalah *landing page* sederhana yang menampilkan nama aplikasi dan deskripsi singkat: "Book Manager adalah sistem digital untuk mengelola, mencari, dan mengunggah koleksi buku." Halaman ini dilengkapi dua tombol utama: *Login* dan *Sign Up*, yang didesain menarik dan responsif dengan bantuan *index.css*, memastikan pengalaman pengguna yang nyaman di berbagai perangkat. Halaman *login.php* adalah halaman autentikasi dengan desain minimalis yang hanya memerlukan *username* dan *password*. Didukung oleh *login.css*, tampilannya simpel tapi profesional. Tombol *Login* jadi fokus utama, dan tersedia tautan ke halaman pendaftaran bagi yang belum punya akun. Halaman *signup.php* difungsikan untuk registrasi pengguna baru melalui formulir yang mencakup data identitas, kontak, dan peran pengguna. Informasi yang dikumpulkan meliputi *username*, nama lengkap, email, kata sandi, jenis kelamin, tanggal lahir, serta peran (siswa, mahasiswa, guru, atau dosen). Fitur unggah foto profil disediakan, baik dari perangkat maupun kamera langsung, dilengkapi dengan

pratinjau gambar. Tombol daftar aktif setelah semua data valid. Desain yang responsif memastikan proses pendaftaran berlangsung efisien dan akurat.

Halaman `Dashboard.php` merupakan pusat kontrol utama dalam sistem *Book Management System*, dirancang untuk memberikan pengalaman yang intuitif dan terorganisir. Ketika pengguna mengakses halaman ini, pengguna disambut dengan tampilan daftar buku yang telah diunggah sebelumnya, serta opsi untuk menambahkan buku baru melalui formulir unggah. Proses unggah *file* pada halaman dashboard dilakukan secara otomatis dengan dukungan teknologi PHP dan Python, memastikan bahwa setiap *file* yang diunggah diproses secara efisien dan akurat. Sebagai bagian dari proses ini, sistem menghasilkan ID unik untuk setiap *file* yang diunggah, seperti `67f5062f2e9c0`. ID unik ini digunakan sebagai nama dasar untuk semua *file* terkait yang dihasilkan selama pemrosesan, termasuk *file* PDF, gambar hasil konversi, *teks* hasil OCR, dan metadata dalam format JSON. Penggunaan ID unik memastikan bahwa setiap *file* memiliki identitas yang konsisten dan dapat dengan mudah dilacak atau dihubungkan satu sama lain. Setelah *file* diunggah, sistem memprosesnya melalui serangkaian langkah otomatis. Jika *file* berformat DOCX, ia dikonversi ke PDF menggunakan skrip Python `docx_to_pdf.py`. File PDF kemudian dikonversi menjadi gambar menggunakan skrip `pdf_to_image.py`, dengan hanya halaman pertama yang diambil untuk efisiensi. Gambar ini diproses menggunakan teknologi OCR (*Optical Character Recognition*) melalui skrip `image_to_text.py` untuk mengekstrak teks.

Teks yang diekstrak kemudian dianalisis menggunakan model NLP (*Natural Language Processing*) dari *OpenAI* melalui skrip `get_metadata.py` untuk mendapatkan metadata penting seperti judul, bahasa, dan topik. Metadata ini disimpan dalam format JSON dengan nama *file* yang juga menggunakan ID unik, seperti `67f5062f2e9c0.json`. Penggunaan ID unik ini memastikan bahwa setiap *file* PDF, gambar, *teks*, dan metadata dapat saling terhubung tanpa kebingungan, bahkan jika ada banyak *file* yang diunggah oleh pengguna yang sama. Setelah semua proses selesai, *file* PDF dan metadata yang dihasilkan disimpan dalam *database*, dan entri baru ditampilkan dalam tabel daftar buku pada halaman dashboard. Setiap entri dalam tabel mencakup judul buku beserta tombol aksi seperti *Lihat*, *Download* dan *Hapus*. Tombol *Lihat* membuka *file* PDF di tab baru, memungkinkan pembacaan langsung terhadap isi buku. Tombol *Download* memfasilitasi pengunduhan *file* PDF ke perangkat lokal, sementara tombol *Hapus* menghapus file dari sistem setelah konfirmasi. Jika belum ada buku yang diunggah, halaman akan menampilkan pesan informatif seperti "*No books found. Please upload a new book,*" yang memberikan panduan jelas tentang langkah selanjutnya.



Gambar 3. Halaman *dashboard user*

Halaman `profile.php` memberikan akses bagi pengguna untuk meninjau dan memperbarui data pribadi yang tersimpan dalam sistem. Informasi yang ditampilkan meliputi nama lengkap, username, email, jenis kelamin, tanggal lahir, peran, serta foto profil (jika tersedia). Jika tidak ada foto, sistem menampilkan pesan informatif sebagai pengganti. Fitur Edit Profile memungkinkan pembaruan data melalui formulir yang telah terisi otomatis dengan informasi terkini. Perubahan disimpan menggunakan metode `POST` dan diproses melalui `query update` pada tabel `users`. Peralihan antara mode tampilan dan edit difasilitasi oleh JavaScript, mendukung interaksi yang intuitif dan dinamis. Desain halaman yang responsif, berkat `profile.css`, memastikan pengalaman pengguna yang efisien dan adaptif di berbagai perangkat.

Halaman `by_titles_initial.php` menyediakan fitur pengelompokan dan pencarian buku berdasarkan huruf awal judul serta kata kunci. Pengguna dapat memilih huruf dari A-Z untuk melihat daftar buku dengan judul yang dimulai dari huruf tersebut. Jika tidak ada buku untuk suatu huruf, sistem menampilkan pesan "Belum Ada *File* Buku." Halaman ini juga menyediakan tabel lengkap A-Z untuk menampilkan distribusi buku berdasarkan huruf awal. Fitur pencarian memungkinkan pengguna mencari buku berdasarkan kata kunci dalam judul. Hasilnya ditampilkan dalam tabel dengan tombol aksi Lihat, *Download* dan Hapus. Tombol Lihat membuka *file* PDF di tab baru, *Download* memfasilitasi pengunduhan *file*, dan Hapus menghapus *file* setelah konfirmasi. Jika tidak ada hasil ditemukan, sistem menampilkan pesan "No results found." Setiap entri buku dilengkapi dengan tombol aksi untuk memudahkan pengguna mengakses atau mengelola *file*. Halaman ini dirancang untuk memberikan pengalaman yang intuitif dan efisien dalam mengelola koleksi buku berdasarkan huruf awal judul atau kata kunci pencarian.



Gambar 4. Halaman berdasarkan huruf awal judul

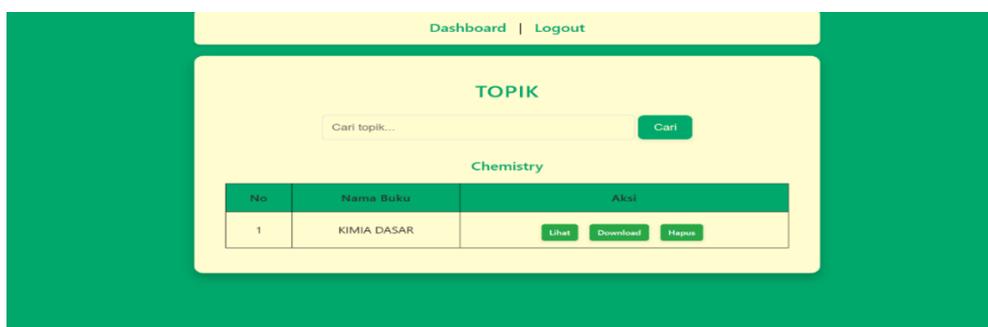
Halaman `by_language.php` memungkinkan pengguna mengelola koleksi buku berdasarkan bahasa. Buku dikelompokkan ke dalam kategori bahasa seperti Inggris, Indonesia, Spanyol, dan lainnya. Jika tidak ada buku untuk suatu bahasa, sistem menampilkan pesan "Tidak ada buku untuk bahasa [nama bahasa]". Fitur pencarian juga disediakan untuk membantu pengguna mencari buku berdasarkan kata kunci dalam judul. Hasil pencarian ditampilkan dalam tabel yang mencantumkan nama *file* dan tindakan yang dapat dilakukan, yaitu Lihat, *Download* dan Hapus. Tombol lihat membuka file PDF di tab baru, *Download* memfasilitasi pengunduhan *file*, dan Hapus menghapus *file* setelah konfirmasi. Jika pencarian tidak menghasilkan hasil, sistem menampilkan pesan "Tidak ada hasil ditemukan". Dengan antarmuka yang intuitif dan fitur pengelompokan berdasarkan bahasa, halaman ini memberikan cara efisien bagi pengguna untuk mengakses dan mengelola koleksi buku. Tombol aksi pada setiap entri memastikan operasi dasar seperti membaca, mengunduh, atau menghapus *file* dapat dilakukan dengan

mudah. Halaman ini mendukung organisasi data yang terstruktur dan responsif dalam sistem.



Gambar 5. Halaman berdasarkan bahasa

Halaman `by_topic.php` membantu pengguna mengelola dan menavigasi koleksi buku berdasarkan topik. Halaman ini menyediakan antarmuka terstruktur untuk melihat daftar buku yang dikelompokkan berdasarkan topik tertentu, serta fitur pencarian untuk menemukan buku berdasarkan kata kunci dalam topik. Pengguna harus *login* terlebih dahulu untuk mengakses halaman ini. Setelah *login*, sistem menampilkan daftar buku berdasarkan topik menggunakan *query SQL* yang mengelompokkan data berdasarkan kolom topik. Jika tidak ada buku untuk suatu topik, sistem menampilkan pesan "Tidak ada buku untuk topik ini." Fitur pencarian memungkinkan pengguna mencari buku dengan memasukkan kata kunci. Hasil pencarian ditampilkan dalam tabel dengan tombol aksi *Lihat*, *Download* dan *Hapus*. Tombol *lihat* membuka *file PDF* di *tab* baru, *Download* memfasilitasi pengunduhan *file*, dan *Hapus* menghapus *file* setelah konfirmasi. Jika tidak ada hasil yang ditemukan, sistem menampilkan pesan "Tidak ada hasil ditemukan." Untuk memastikan identifikasi unik, setiap *file* diberi ID unik yang digunakan sebagai nama dasar untuk *file PDF*, gambar hasil konversi, *teks* hasil OCR, dan metadata dalam format JSON. Penggunaan ID unik ini memastikan bahwa *file-file* terkait tetap dapat dibedakan meskipun memiliki judul atau topik yang sama.



Gambar 6. Halaman berdasarkan topik

4. KESIMPULAN DAN SARAN

Kesimpulan

Sistem pengelolaan buku digital ini berhasil dirancang dan diimplementasikan sebagai solusi yang efisien untuk mengorganisir, mencari, dan mengelola koleksi buku secara digital. Dengan integrasi teknologi *Natural Language Processing* (NLP), sistem mampu mengekstraksi metadata penting seperti judul, bahasa, dan topik dari konten buku secara otomatis. Fitur ini memungkinkan pengguna untuk mengelompokkan dan mencari buku berdasarkan berbagai kriteria, seperti huruf awal judul, bahasa, atau topik, dengan cepat dan akurat. Selain itu, sistem juga dilengkapi dengan fitur unggah *file*, konversi otomatis (DOCX ke PDF), ekstraksi *teks* menggunakan OCR, serta manajemen data pengguna melalui *database* MySQL. Struktur folder yang modular memisahkan aset statis, *file* unggahan, skrip Python, dan *file* PHP utama, sehingga memastikan kemudahan dalam pengembangan dan pemeliharaan sistem. Antarmuka pengguna yang responsif dan desain CSS yang konsisten memberikan pengalaman yang intuitif dan nyaman bagi pengguna. Fitur-fitur seperti autentikasi pengguna, *edit* profil, serta tombol aksi lihat, *download* dan hapus pada setiap entri buku menunjukkan bahwa sistem ini dirancang dengan fokus pada kepraktisan dan kemudahan penggunaan. Keberhasilan implementasi Model NLP sebagai inti analisis konten membuktikan bahwa sistem ini tidak hanya memenuhi kebutuhan pengguna saat ini tetapi juga memiliki potensi untuk dikembangkan lebih lanjut. Dengan pendekatan modular dan integrasi teknologi modern, aplikasi ini menawarkan skalabilitas yang baik untuk peningkatan fungsionalitas di masa mendatang. Secara keseluruhan, sistem ini berhasil menjadi alat pengelolaan buku digital yang praktis, efisien, dan inovatif, menjawab tantangan dalam pengorganisasian koleksi buku secara digital.

Saran

Untuk pengembangan lebih lanjut, disarankan agar sistem ini ditingkatkan dengan menambahkan fitur-fitur tambahan seperti rekomendasi buku berdasarkan preferensi pengguna atau integrasi dengan *API* eksternal untuk mengakses metadata buku dari sumber global seperti *Google Books*. Selain itu, sistem dapat diperluas dengan menambahkan dukungan untuk format *file* lain seperti EPUB atau MOBI, serta meningkatkan kemampuan analisis NLP untuk menghasilkan metadata yang lebih rinci, seperti ringkasan otomatis atau klasifikasi genre yang lebih spesifik. Dari segi keamanan, disarankan untuk menerapkan validasi input yang lebih ketat dan enkripsi *password* menggunakan metode *hashing* yang lebih kuat, seperti *bcrypt*. Terakhir, pengujian lebih lanjut terhadap performa aplikasi pada skala besar akan membantu memastikan bahwa sistem tetap stabil dan responsif meskipun digunakan oleh banyak pengguna secara bersamaan.

5. DAFTAR PUSTAKA

- [1] M. Atika and R. Sayekti, "Open Access under Creative Commons Attribution Non Commercial Share Alike 4.0 International License (CC-BY-NC-SA) Studi Literatur Review Sistem Informasi Perpustakaan Berbasis Artificial Intelligence (AI) Library Information System Based on Artificial Intelligence (AI): Literatur Review," *Palimpsest: Jurnal Ilmu Informasi dan Perpustakaan*, vol. 14, p. 2023, doi:10.20473/pjil.v14i1.4645.
- [2] I. P. B. Kusuma, L. P. C. Prabandari, M. P. Restami, "Sistem Informasi Pengarsipan Surat Pertanggung Jawaban Berbasis Web Badan Penanggulangan

- Bencana Daerah Buleleng,” *Jurnal Teknologi Informasi dan Komputer*, vol. 9, no. 4, pp. 479–490, 2023.
- [3] F. D. Dermawan and Supriyono, “Implementasi Artificial Intelligence dalam Pembuatan Website Klasifikasi Genre Buku,” *bit-Tech*, vol. 7, no. 2, pp. 618–627, Dec. 2024, doi: 10.32877/bt.v7i2.1977.
- [4] M. Yusuf and S. T. Agama, “Pustakaloka Jurnal Kajian Informasi dan Perpustakaan Corresponding Author: Peran Artificial Intelligence (AI) sebagai Pendukung Otomatisasi Perpustakaan,” vol. 16, 2024, doi: 10.21154/pustakaloka.v16i1.7516.
- [5] O. Y. Frinaldi Aldri, “Aplikasi Isumbarmambaco: Inovasi Budaya Kerja dalam Pelayanan Publik Berbasis Digital di Dinas Kearsipan dan Perpustakaan Provinsi Sumatra Barat,” *Journal of Education Research*, vol. 5, no. 4, pp. 5874–5885, 2024.
- [6] F. Arifin, H. Sibyan, and N. Hasanah, “Rancang Bangun Chatbot Pada Sistem Ekapta Berbasis Natural Language Processing Dengan Algoritma Artificial Neural Network,” vol. 4, no. 1, 2025, [Online]. Available: <https://ojs.unsiq.ac.id/index.php/biner>
- [7] J. Ilmu Perpustakaan dan, M. Farid Wajdi, and M. Isra Hajiri, “TIK Ilmeu Tantangan Adaptasi Kecerdasan Buatan dalam Layanan Perpustakaan Perguruan Tinggi,” vol. 8, no. 2, pp.2580–3662, 2024, doi: 10.29240/tik.v8i2.10901.
- [8] R. Maolina Maora, R. Mulyati, and D. Nugraha, “Cipasung Techno Pesantren Scientific Journal Penerapan Natural Language Processing dan Convolutional Neural Network dalam Pengembangan Asisten Pertanian Berbasis Chatbot,” vol. 19, no. 1, 2025.
- [9] M. Yusuf and S. T. Agama, “Pustakaloka Jurnal Kajian Informasi dan Perpustakaan Corresponding Author: Peran Artificial Intelligence (AI) sebagai Pendukung Otomatisasi Perpustakaan,” vol. 16, 2024, doi: 10.21154/pustakaloka.v16i1.7516.
- [10] M. I. Mubarok and M. Abdi, “Implementasi Natural Language Processing Dalam Perancangan Aplikasi Chatbot Pada Fikti Umsu,” 2024.