

Rancang Bangun Program Layanan Konsultasi Bantuan Hukum Berbasis Web (Studi Kasus: Sumba Barat)

Florentina Bili¹, Antar Maramba Jawa², Alexander Talo Popo³

^{1,2,3} Teknik Informatika, Universitas Stella Maris Sumba

E-mail : ¹biliflorentina@gmail.com, ²andjawa89@gmail.com, ³alexpopo484@gmail.com

KEYWORDS:

Legal Aid Consulting Services, Natural Language Toolkit.

ABSTRACT

Web-based legal consulting services are increasingly needed in today's digital era, especially to facilitate public access to legal information. This research aims to design and implement a web-based legal consultation service system by utilizing the Natural Language Toolkit (NLTK) method to analyze and understand legal texts. The NLTK method is used to perform natural language processing (NLP) to identify patterns and important information in legal texts, and provide recommendations based on user input. This system is expected to help users, and provide recommendations based on user input. This system is expected to help users, especially those who are unfamiliar with the law, to get relevant and timely legal explanations, especially in the field of land law. The results of the application testing show that the use of NLTK is able to read and display user requests in providing legal consultation automatically. Thus, this system can be an innovative and practical solution in supporting legal services that are more accessible to the wider community.

KATA KUNCI:

Layanan Konsultasi Bantuan Hukum, Natural Language Toolkit.

ABSTRAK

Layanan konsultasi hukum berbasis web semakin diperlukan dalam era digital saat ini, terutama untuk memudahkan akses masyarakat terhadap informasi hukum. Penelitian ini bertujuan untuk merancang dan mengimplementasikan sebuah sistem layanan konsultasi hukum berbasis web dengan memanfaatkan metode Natural Language Toolkit (NLTK) untuk menganalisis dan memahami teks hukum. Metode NLTK digunakan untuk melakukan pemrosesan bahasa alami (Natural Language Processing/NLP) guna mengidentifikasi pola-pola dan informasi penting dalam teks hukum, serta memberikan rekomendasi berdasarkan input pengguna. Sistem ini diharapkan dapat membantu pengguna, khususnya yang awam terhadap hukum untuk mendapatkan penjelasan hukum yang relevan dan tepat waktu terutama dalam bidang hukum pertanahan. Hasil testing aplikasi menunjukkan bahwa penggunaan NLTK mampu membaca dan menampilkan permintaan user dalam pemberian konsultasi hukum secara otomatis. Dengan demikian, sistem ini dapat menjadi solusi yang inovatif dan praktis dalam mendukung layanan hukum yang lebih mudah diakses oleh masyarakat luas.

PENDAHULUAN

Bantuan hukum tersedia bagi semua orang untuk membantu mendapatkan perlakuan hukum yang sama ketika tidak menerima perlakuan yang layak sesuai dengan ketentuan hukum yang berlaku[1]. Peraturan dan terminologi hukum yang kompleks dapat menimbulkan tantangan bagi individu yang kurang beruntung dan mereka yang memiliki pengetahuan hukum terbatas dalam proses mencari keadilan[2]. Hal ini membuat Lembaga Bantuan Hukum (LBH) Sarneli Waikabubak menginginkan masyarakat yang kurang mampu dan masyarakat yang memiliki pengetahuan terbatas akan hukum dapat memperoleh pengetahuan hukum melalui konsultasi[3].

Saat ini untuk memperoleh bantuan hukum, masyarakat dapat mengakses layanan yang disediakan oleh Lembaga Bantuan Hukum (LBH) Sarneli Waikabubak dengan mendatangi kantor institusi[4]. Sementara itu, kesibukan masyarakat dalam aktivitas sehari-hari dan minimnya waktu luang membuat mereka tidak mungkin memiliki kesempatan untuk mengunjungi lembaga bantuan hukum atau kantor pengacara. Selain itu, kehadiran kantor LBH Sarneli Waikabubak ini tidak dikenal luas karena memiliki jangkauan pasar yang terbatas[5]. Untuk memberikan layanan hukum kepada masyarakat, LBH Sarneli Waikabubak harus menyediakan layanan konsultasi

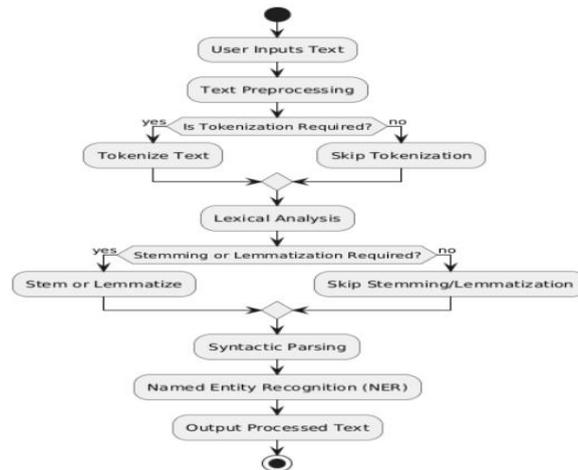
hukum secara online agar masyarakat yang kurang mampu dapat mengakses pengetahuan hukum yang dibutuhkan oleh masyarakat. Teknologi informasi dapat memudahkan masyarakat untuk mengakses informasi kapan dan dimana saja[6].

Dalam penelitian ini, peneliti melakukan rekayasa layanan hukum berbasis web khususnya terkait sengketa tanah. Website ini memiliki kemampuan untuk menyajikan informasi hukum yang komprehensif dan mendetail berdasarkan data resmi yang diakui oleh lembaga-lembaga terkait[7]. Dengan pendekatan yang sistematis, layanan ini menjabarkan aturan-aturan hukum yang relevan secara jelas dan terstruktur, sehingga mudah dipahami oleh pengguna awam maupun profesional di bidang hukum. Setiap saran hukum yang diberikan dapat disesuaikan dengan kasus yang dihadapi oleh pengguna dengan mengacu pada dokumen-dokumen atau regulasi hukum terkini. Salah satu keunggulan dari layanan ini adalah kemampuan untuk menghasilkan dokumen penjelasan hukum dalam format PDF[8]. Dokumen ini dapat diunduh atau dicetak oleh pengguna yang memungkinkan mereka untuk memiliki referensi hukum yang dapat dipelajari lebih lanjut atau digunakan sebagai dokumen pendukung dalam proses penyelesaian sengketa tanah. Dengan tampilan yang mudah diakses melalui antarmuka web, layanan ini berfungsi sebagai solusi yang efisien dan efektif bagi masyarakat yang membutuhkan pemahaman hukum secara cepat tanpa harus mengunjungi kantor hukum secara fisik[9]. Metode yang digunakan dalam penelitian ini adalah metode *Natural Language Toolkit (NLTK)* untuk menganalisis dan memahami teks hukum. Metode *NLTK* digunakan untuk melakukan pemrosesan bahasa alami (*Natural Language Processing/NLP*) guna mengidentifikasi pola-pola dan informasi penting dalam teks hukum, serta memberikan rekomendasi berdasarkan input pengguna. Sistem ini diharapkan dapat membantu pengguna, khususnya yang awam terhadap hukum, untuk mendapatkan penjelasan hukum yang relevan dan tepat waktu, terutama dalam bidang hukum pertanahan.diberikan[10].

Dengan adanya layanan hukum berbasis web khususnya dalam masalah sengketa tanah diharapkan dapat membantu masyarakat Sumba (Sumba Barat) untuk mendapatkan gambaran hukum terkait kasus hukum (sengketa tanah) yang dihadapi. Berdasarkan latar belakang di atas, maka peneliti melakukan penelitian yang berjudul “Rancang Bangun Program Layanan Konsultasi Bantuan Hukum Berbasis Web (Studi Kasus: Kabupaten Sumba Barat)”.

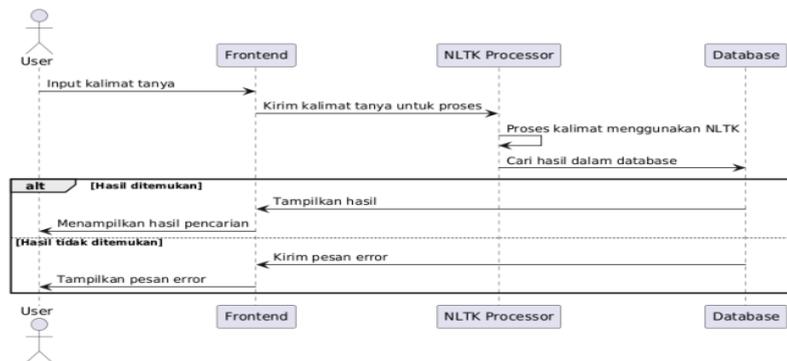
METODE PENELITIAN

Metode yang digunakan dalam penelitian ini adalah metode *Natural Language Toolkit (NLTK)* untuk menganalisis dan memahami teks hukum. Metode *NLTK* digunakan untuk melakukan pemrosesan bahasa alami (*Natural Language Processing/NLP*) guna mengidentifikasi pola-pola dan informasi penting dalam teks hukum, serta memberikan rekomendasi berdasarkan input pengguna. Sistem ini diharapkan dapat membantu pengguna, khususnya yang awam terhadap hukum, untuk mendapatkan penjelasan hukum yang relevan dan tepat waktu, terutama dalam bidang hukum pertanahan.diberikan. Berikut gambaran *NLTK Flow* :



Gambar 1 NLTK Flow

NLTK Flow integrasi dengan proyek saat ini :



Gambar 2. NLTK Flow integrasi

Penjelasan :

1. *User* : Menginputkan kalimat tanya.
2. *Frontend (FE)* : Menerima input dan mengirimkan kalimat tersebut ke komponen pemrosesan *NLP*.
3. *NLTK Processor (NLP)* : Mengolah kalimat menggunakan *NLTK* untuk mendapatkan informasi yang diperlukan.
4. *Database (DB)* : Setelah pemrosesan *NLP* selesai, data dicari dalam *database*.
5. Jika hasil ditemukan, maka hasil ditampilkan ke *user*.
6. Jika tidak ditemukan, pesan error dikirim dan ditampilkan.

Fitur Utama :

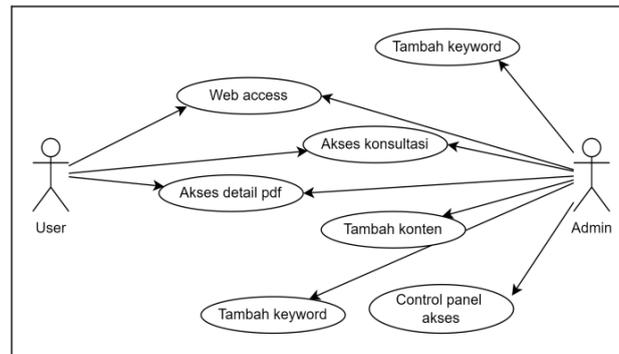
1. *Tokenisasi* : Memecah teks menjadi kata atau kalimat.
2. *Stemming* : Mengubah kata ke bentuk dasar (root).
3. *Lemmatization* : Mirip dengan stemming, tetapi mempertahankan makna linguistik.
4. *Part-of-Speech Tagging* : Menandai kata dengan jenisnya (kata benda, kata kerja, dsb.).

5. *Named Entity Recognition (NER)*: Mendeteksi entitas penting seperti nama orang, tempat, dan organisasi.
6. *Parsing*: Menganalisis struktur gramatikal kalimat.
7. *Stopword Removal*: Menghapus kata-kata umum yang tidak relevan (seperti "dan", "atau", "adalah").

HASIL DAN PEMBAHASAN

1. Use Case Diagram

Salah satu jenis diagram *UML (Unified Modeling Language)* yang menunjukkan fungsionalitas, luas, dan antarmuka pengguna suatu sistem adalah diagram kasus penggunaan. Diagram kasus penggunaan menunjukkan tindakan terperinci yang dapat dilakukan aktor pada kasus penggunaan serta interaksi antara pengguna (aktor) dan sistem (kasus penggunaan). Gambar *Use Case Diagram* dapat dilihat pada gambar berikut :



Gambar 4. Use Case Diagram

1. Penjelasan Aktor :

User : Pengguna umum yang menggunakan layanan konsultasi hukum.

Admin : Pengguna dengan hak akses administratif yang bertanggung jawab atas pengelolaan konten dalam sistem.

2. Use Case dan Relasinya

a. Aktor : *User*

- i. *Web Access* : User dapat mengakses halaman web layanan untuk menggunakan fitur-fitur yang tersedia.
- ii. *Akses konsultasi* : User dapat mengakses layanan konsultasi hukum terkait sengketa tanah. Ini memungkinkan mereka mendapatkan penjelasan hukum yang relevan berdasarkan kebutuhan mereka.
- iii. *Akses detail PDF* : Setelah menerima penjelasan hukum, user dapat melihat detail penjelasan tersebut dalam format *PDF* yang dapat diunduh dan digunakan lebih lanjut sebagai dokumen referensi.

- b. Aktor: Admin
 - i. *Web Access* : Admin juga memiliki kemampuan untuk mengakses layanan melalui *web* seperti user, tetapi dengan hak istimewa tambahan.
 - ii. Tambah konten : Admin bertanggung jawab menambah konten yang relevan dengan konsultasi hukum, termasuk mengunggah penjelasan baru, pasal, dan informasi hukum terkait lainnya.
 - iii. Tambah *keyword*: Admin dapat menambah kata kunci (*keyword*) yang terkait dengan konten hukum, yang memudahkan pencarian bagi user.
 - iv. Control panel akses: Admin memiliki akses ke *control panel* untuk mengelola layanan secara keseluruhan, mengawasi aktivitas *user*, serta melakukan perubahan dalam sistem.

- a. Interaksi Antara Aktor dan Use Case
 - i. *User* : Akses awal dimulai dengan *Web Access* yang memberikan mereka kemampuan untuk menggunakan layanan utama seperti Akses konsultasi dan Akses detail *PDF*. Dengan mengakses konsultasi, user mendapatkan informasi hukum yang disediakan oleh sistem, yang kemudian dapat diunduh dalam format *PDF* jika mereka ingin melihat detail lebih lanjut.
 - ii. Admin : Selain fitur yang bisa diakses oleh *user*, admin memiliki peran tambahan untuk mengelola konten. Admin dapat menambah konten konsultasi hukum yang tersedia melalui use case Tambah konten, serta menambahkan kata kunci yang relevan dengan konten tersebut melalui tambah keyword. Admin juga mengontrol sistem melalui control panel akses, yang merupakan fitur eksklusif bagi admin untuk memodifikasi dan memantau aktivitas dalam layanan.

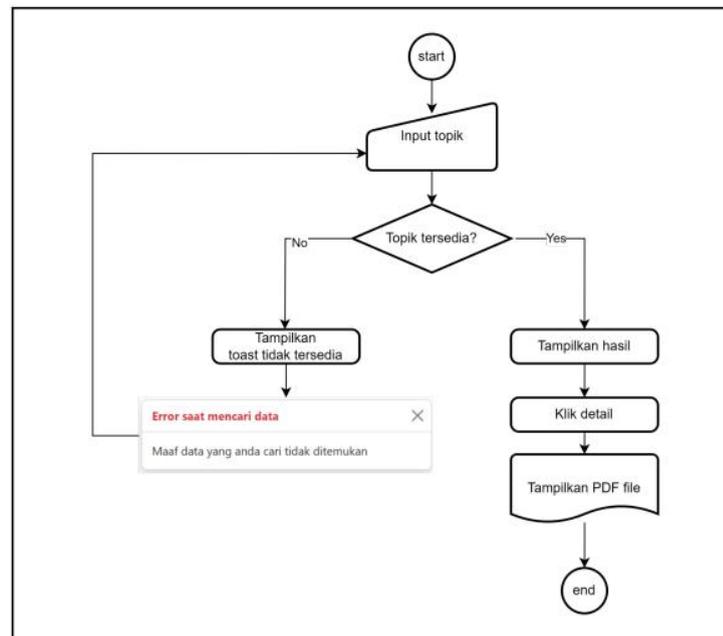
Kesimpulan :

User memiliki akses untuk menggunakan layanan konsultasi hukum, melihat penjelasan hukum, dan mengunduh detail penjelasan tersebut dalam bentuk *PDF*. Admin memiliki peran lebih aktif, yaitu selain dapat mengakses layanan konsultasi seperti user, mereka juga bertanggung jawab dalam menambah konten baru, mengelola kata kunci, dan memantau keseluruhan sistem melalui *control panel*.

2. Activity Diagram

Diagram aktivitas adalah sejenis diagram *UML (Unified Modeling Language)* yang digunakan untuk menjelaskan alur kerja atau proses perangkat lunak atau proses bisnis. Analisis perangkat lunak dan fase desain sering menggunakan diagram ini. Activity diagram dapat dilihat pada gambar berikut :

A. User Cari Topik



Gambar 5. Activity diagram

Gambar diatas menampilkan activity diagram yang menjelaskan alur kerja dari proses pencarian topik dalam sebuah sistem layanan hukum berbasis web. Berikut penjelasan rinci dari tiap langkah dalam diagram tersebut :

a. Start

Proses dimulai di titik ini, mengindikasikan bahwa pengguna telah mengakses layanan dan siap memulai pencarian topik hukum yang diinginkan.

b. Input Topik

Pengguna memasukkan kata kunci atau topik yang ingin dicari. Ini adalah langkah input di mana pengguna mengisi informasi mengenai topik hukum yang mereka butuhkan.

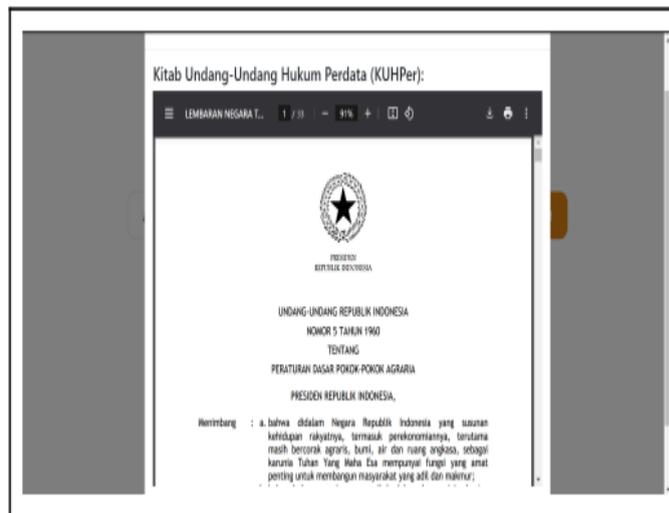
c. Keputusan: "Topik Tersedia?"

Sistem kemudian memeriksa apakah topik yang dimasukkan pengguna tersedia di dalam basis data: Jika topik tersedia (Yes) : Sistem akan menampilkan hasil pencarian kepada pengguna. Jika topik tidak tersedia (No) : Sistem akan menampilkan pesan kesalahan (error).

d. Alur "No" (Topik Tidak Tersedia) Tampilkan toast tidak tersedia :

Jika topik yang dicari tidak ditemukan di dalam basis data, system akan menampilkan pesan kesalahan berupa toast notification. Pesan tersebut menyampaikan bahwa "Maaf data yang Anda cari tidak ditemukan". Contoh pesan error yang muncul: "Error saat mencari data: Maaf data yang anda cari tidak ditemukan". Pesan ini memberikan informasi kepada pengguna bahwa data yang mereka cari tidak ada dalam sistem. Setelah pesan kesalahan ditampilkan, pengguna akan kembali ke langkah awal untuk mencoba mencari topik lain.

- e. Alur "Yes" (Topik Tersedia) Tampilkan hasil :
Jika topik tersedia, sistem akan menampilkan hasil pencarian yang relevan dengan topik yang dimasukkan.
- f. Klik detail :
Pengguna kemudian dapat memilih untuk melihat informasi lebih lanjut dengan mengklik opsi "detail".
- g. Tampilkan PDF file :
Setelah pengguna mengklik detail, system akan menampilkan dokumen dalam bentuk PDF yang berisi informasi hukum lebih lanjut terkait topik tersebut. PDF ini bisa diunduh dan digunakan oleh pengguna untuk referensi lebih lanjut.
- h. End :
Proses selesai setelah file PDF ditampilkan.

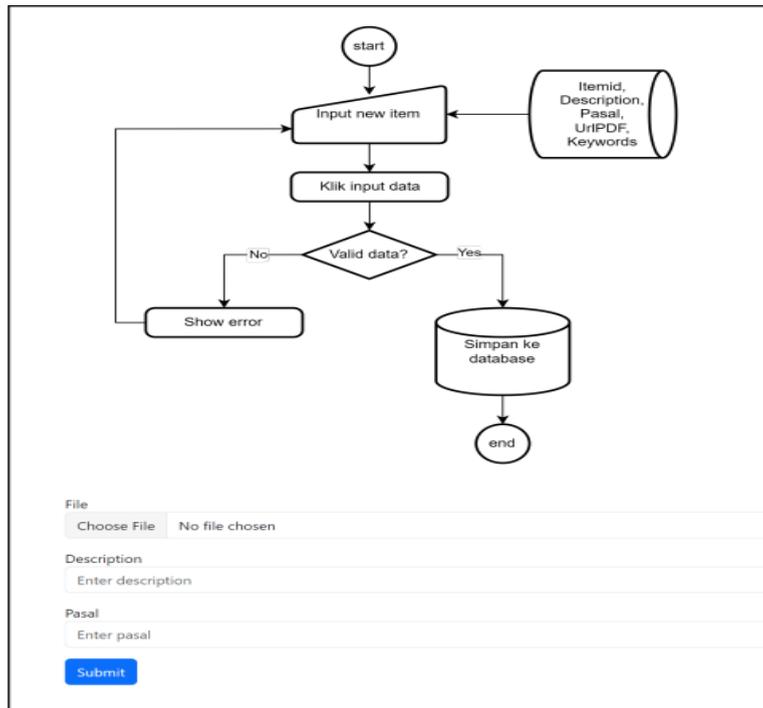


Gambar 6. Tampilan PDF

KESIMPULAN

Diagram aktivitas ini menunjukkan alur pencarian topik di dalam layanan konsultasi hukum berbasis web. Alurnya cukup sederhana: Pengguna memulai dengan memasukkan topik yang ingin mereka cari. Sistem memeriksa ketersediaan topik. Jika topik tersedia, hasil akan ditampilkan, dan pengguna dapat mengakses dokumen PDF yang relevan. Jika topik tidak tersedia, sistem akan menginformasikan kepada pengguna melalui pesan kesalahan. Proses ini memungkinkan pengguna untuk mendapatkan informasi hukum yang mereka butuhkan dengan mudah, serta memberikan umpan balik yang jelas ketika topik yang dicari tidak ditemukan.

B. Admin Tambah topik



Gambar 7. Aktiviti Diagram

Gambar diatas menampilkan aktivitas admin dalam menambah topik kasus hukum yang menjelaskan alur kerja dari proses penambahan topik dalam sebuah sistem layanan hukum berbasis web. Berikut penjelasan rinci dari tiap langkah dalam diagram tersebut :

Star

Proses dimulai di titik ini, mengindikasikan bahwa admin telah mengakses layanan dan siap memulai penambahan topik hukum yang diinginkan.

a. Input New Item

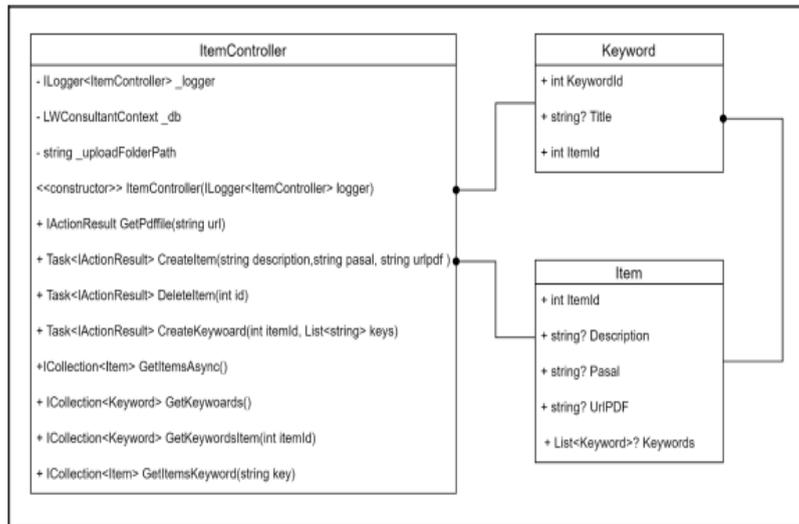
Admin menambahkan item baru yang ingin ditambahkan ke dalam database dalam bentuk pdf.

b. Validasi Data

Sistem melakukann validasi data berdasarkan item baru yang diinput kedalam database. Jika terjadi kesalahan penginputan maka system akan menampilkan notifikasi error dan jika tidak terjadi kesalahan maka system akan menampilkan hasil sukses.

3. Class Diagram

Class diagram menunjukkan bagaimana skema arsitektur sebuah sistem yang sedang dirancang, serta class diagram juga digunakan untuk menunjukkan interaksi antar class di dalam sistem. Adapun class diagram usulan hasil normalisasi ditunjukkan pada gambar berikut :

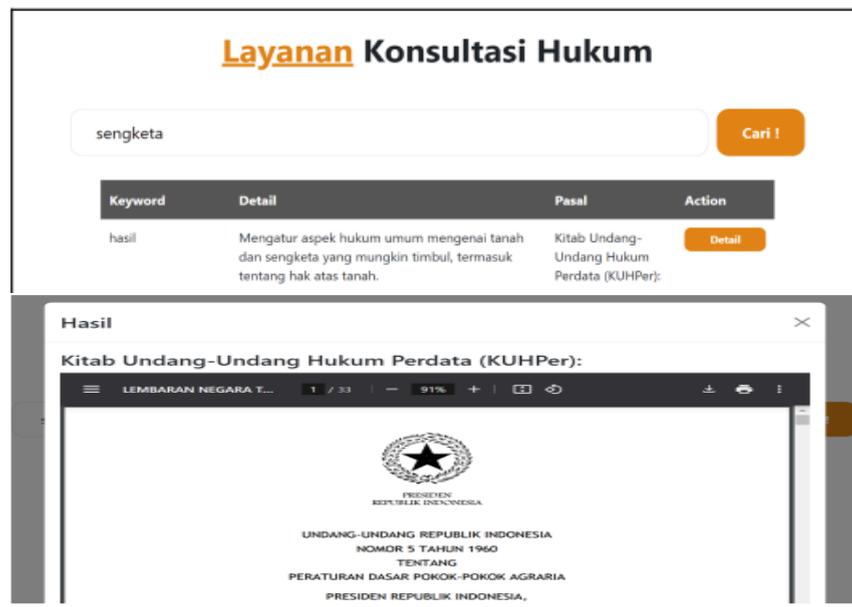


Gambar 8. Class Diagram

4. Mockup wireframe UI

A. User Mencari Topik dan Data Ditemukan

User dapat mencari topik hukum yang sesuai dengan kasus hukum yang dihadapi. Jika topik tersedia, sistem akan menampilkan hasil pencarian yang relevan dengan topik yang dimasukkan. Berikut gambar topik dalam layanan konsultasi hukum :

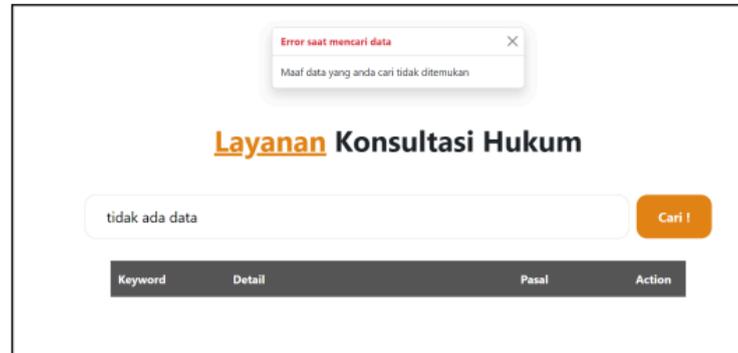


Gambar 9. User Mencari Topik

B. User Mencari Topik dan Data Tidak Ditemukan

Jika topik yang dicari tidak ditemukan di dalam basis data, system akan menampilkan pesan kesalahan berupa toast notification. Pesan tersebut menyampaikan bahwa "Maaf data yang Anda cari tidak ditemukan". Contoh pesan error yang muncul: "Error saat mencari data: Maaf data yang

anda cari tidak ditemukan". Pesan ini memberikan informasi kepada pengguna bahwa data yang mereka cari tidak ada dalam sistem. Setelah pesan kesalahan ditampilkan, pengguna akan kembali ke langkah awal untuk mencoba mencari topik lain. Gambar mencari topik dan data tidak ditemukan dapat dilihat pada gambar berikut :



Gambar 10. Data Tidak Ditemukan

C. Admin input data baru

Gambar di bawah menampilkan aktivitas admin dalam menambah data baru yang berkaitan dengan kasus hukum yang menjelaskan alur kerja dari proses penambahan topik dalam sebuah sistem layanan hukum berbasis web.

The image shows an admin form for adding new data. It has three main sections: "File" with a "Choose File" button and the text "UU Nomor 5 Tahun 1960.pdf"; "Description" with the text "Sengketa tanah"; and "Pasal" with the text "Pasal 9 ayat 1". There is a blue "Submit" button at the bottom left.

Gambar 11. Admin Tambah Data Baru

- D. Source code
 - 1. Controller

Berikut gambar source code controller

```

using Microsoft.AspNetCore.Mvc;
using System;
using System.Linq;
using LWConsultantSQL;
using System.Data.Common;
using Microsoft.EntityFrameworkCore.Metadata.Internal;
using Microsoft.EntityFrameworkCore;
namespace LWController.Controllers;

[ApiController]
[Route("[controller]")]

public class ItemController: ControllerBase{
    private readonly ILogger<ItemController> _logger;
    private readonly LWConsultantContext _db;
    private readonly string _uploadFolderPath = "Upload";

    public ItemController(ILogger<ItemController> logger)
    {
        _db = new LWConsultantContext();
    }

    _db.Database.Migrate();
    _logger = logger;
}

[HttpPost("AddPdfFile")]
public async Task<IActionResult> AddPdfFile(IFormFile
file) {
    if (file == null || file.Length == 0)
    {
        return BadRequest("No file was uploaded.");
    }
    if (!Directory.Exists(_uploadFolderPath))
    {
        Directory.CreateDirectory(_uploadFolderPath);
    }
    var filePath = Path.Combine(_uploadFolderPath,
file.FileName);
    using (var stream = new FileStream(filePath,
 FileMode.Create))
    {
        await file.CopyToAsync(stream);
    }
    return Ok(new { Message = "File uploaded
successfully.", fileName = file.FileName });
}

[HttpGet("getpdffile/{url}")]
public IActionResult GetPdfFile(string url) {
    if (System.IO.File.Exists(url))
    {
        var fileBytes = System.IO.File.ReadAllBytes(url);
        return File(fileBytes, "application/pdf", url);
    }
    else
    {
}

[HttpPost("CreateItem")]
public async Task<IActionResult> CreateItem(string
description, string pasal, string urlpdf ){
    Item item = new Item{
        Description = description,
        Pasal = pasal,
        UrlPDF = urlpdf
    };
    _db.Items.Add(item);
    await _db.SaveChangesAsync();
    return Ok(item);
}

[HttpDelete("DeleteItem")]
public async Task<IActionResult> DeleteItem(int id) {
    var item = await _db.Items.FindAsync(id);
    if (item == null)
    {
        return BadRequest("item not found");
    }
    _db.Items.Remove(item);
    await _db.SaveChangesAsync();
    return Ok("deleted successfully");
}

[HttpPost("CreateKeyword")]
public async Task<IActionResult> CreateKeyword(int
itemId, List<string> keys){
    var item = _db.Items.Where(b => b.ItemId == itemId
).First();
    foreach(var key in keys){
        item.Keywords!.Add(new Keyword{
            Title = key
        });
    }
}

```

```

        await _db.SaveChangesAsync();

        return Ok();
    }

    [HttpGet("GetItems")]
    public ICollection<Item> GetItemsAsync(){
        var items = _db.Items.ToList();
        return items;
    }

    [HttpGet("GetKeywords")]
    public ICollection<Keyword> GetKeywords(){
        var keys = _db.Keywords.ToList();
        return keys;
    }

    [HttpGet("GetKeywordsItem")]
    public ICollection<Keyword> GetKeywordsItem(int itemId){
        var keys = _db.Keywords.Where(item => item.ItemId
        ==itemId);
        return keys.ToList();
    }

    [HttpGet("GetItemsKeyword")]
    public ICollection<Item> GetItemsKeyword(string key){
        var items = _db.Items.Where(item =>
        item.Keywords!.Any(kw =>
        kw.Title!.ToLower().Contains(key.ToLower())));
        return items.ToList();
    }
}

```

Gambar 11. Source Code Controller

2. Model Item

Berikut gambar model item :

```

#region Assembly LWConsultantSQL, Version=1.0.0.0,
Culture=neutral, PublicKeyToken=null
//
C:\Users\Formulatrix\Documents\project\4\LWConsultantSQL\obj\
Debug\net8.0\ref\LWConsultantSQL.dll

```

```

#endregion

#nullable enable

using System.Collections.Generic;

namespace LWConsultantSQL
{
    public class Item
    {
        public Item();

        public int ItemId { get; set; }
        public string? Description { get; set; }
        public string? Pasal { get; set; }
        public string? UrlPDF { get; set; }
        public List<Keyword>? Keywords { get; }
    }
}

```

Gambar 12. Model Item

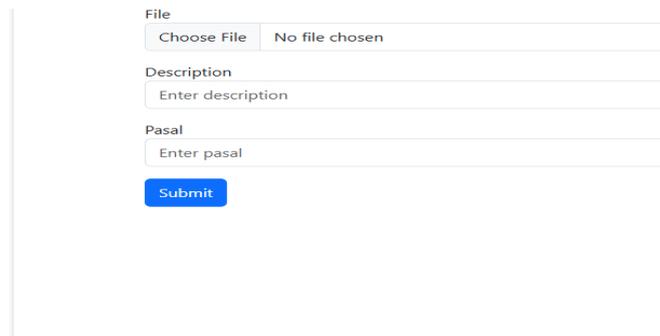
Hasil Akhir Dari Aplikasi

1. User



Gambar 13. Hasil Akhir User

2. Admin



Gambar 14. Hasil Akhir Admin

Pengujian

Pada penelitian ini menggunakan metode pengujian black box. Pengujian Black Box adalah metode pengujian perangkat lunak di mana penguji mengevaluasi fungsi perangkat lunak. Fokus utama dalam pengujian ini adalah pada input dan output sistem sesuai dengan spesifikasi atau kebutuhan yang telah ditentukan

a. Pengujian Black Box untuk user

Id	Skenario	Input/aksi	Ekspektasi	Hasil
1	Input topik valid	input topik "Hukum tanah"	Data ditemukan dan ditampilkan pada list	Berhasil
2	Input topik tidak valid	Input topik "Lorem ipsum"	Data tidak ditemukan, error toast message ditampilkan	Berhasil

3	Klik detail hasil pencarian	Jalankan test ID 1 dan klik detail pada list yang ditampilkan	Tampilan akan diarahkan ke detail PDF sesuai dengan topik yang dipilih	Berhasil
4	Akses web layanan konsultasi	Masukan alamat website dan akses halaman tersebut	Halaman akan diarahkan ke halaman user atau halaman pengisian topik	Berhasil
5	Akses PDF file	Jalankan test ID 3 dan akses tampilan PDF	PDF yang memiliki relasi dengan topik yang dipilih akan ditampilkan dan user dapat mengaksesnya	Berhasil

b. Pengujian Black Box untuk admin

Id	Skenario	Input/aksi	Ekspektasi	Hasil
1	Tambah topik baru dengan data valid	Akses halaman admin dan isi semua data dengan data yang valid kemudian klik submit	Data baru akan ditambahkan pada database	Berhasil
2	Tambah topik baru dengan data yang tidak valid	Akses halaman admin dan isi beberapa data dan biarkan input data yang lain menjadi kosong kemudian klik submit	Admin akan diberikan peringatan untuk setiap fields yang bermasalah (data kosong, tidak valid, atau error)	Berhasil
3	Akses halaman admin	Masukan alamat website dan akses	User akan diarahkan ke halaman	Berhasil

		halaman tersebut	admin	
--	--	---------------------	-------	--

KESIMPULAN

Penelitian ini berhasil mengembangkan layanan konsultasi hukum berbasis web dengan menggunakan metode NLTK untuk pemrosesan bahasa alami dalam teks hukum. Implementasi metode NLTK terbukti efektif dalam menganalisis teks hukum dan memberikan rekomendasi yang relevan berdasarkan pertanyaan pengguna. Sistem ini tidak hanya memberikan kemudahan akses informasi hukum, tetapi juga meningkatkan efisiensi dalam proses konsultasi, khususnya dalam kasus-kasus hukum pertanahan. Pengujian menunjukkan bahwa sistem ini mampu memberikan hasil yang akurat dan responsif terhadap input pengguna, sehingga dapat membantu masyarakat dalam memahami dan mendapatkan solusi hukum secara lebih cepat dan mudah. Penggunaan teknologi seperti NLTK dalam layanan hukum berbasis web berpotensi untuk terus dikembangkan guna mendukung transformasi digital dalam bidang hukum di masa depan.

DAFTAR PUSTAKA

- [1] D. M. Listianingrum, B. Budiharto, and S. Mahmudah, "PERLINDUNGAN HUKUM TERHADAP KONSUMEN DALAM PELAYANAN KESEHATAN BERBASIS APLIKASI ONLINE," *Diponegoro Law J.*, vol. 8, no. 3, pp. 1889–1904, 2019.
- [2] A. Zaini, M. Dianto, and R. R. Mulyani, "Pentingnya penggunaan media bimbingan dan konseling dalam layanan informasi," in *Prosiding Seminar Nasional Bimbingan Dan Konseling Universitas Negeri Malang*, 2020, pp. 126–131.
- [3] D. Muriyatmoko, B. Sholeh, and S. N. Utama, "Rancang Bangun Sistem Informasi Layanan Konsultasi Bantuan Hukum LPKBH Al-Baihaqy Surabaya," *J. Inform. Polinema*, vol. 7, no. 1, pp. 7–16, 2020.
- [4] S. R. Yulistina, T. Nurmala, R. M. A. T. Supriawan, S. H. I. Juni, and A. Saifudin, "Penerapan Teknik Boundary Value Analysis untuk Pengujian Aplikasi Penjualan Menggunakan Metode Black Box Testing," *J. Inform. Univ. Pamulang*, vol. 5, no. 2, pp. 129–135, 2020.
- [5] B. Manan, A. Abdurahman, and M. Susanto, "Pembangunan Hukum Nasional Yang Religius: Konsepsi Dan Tantangan Dalam Negara Berdasarkan Pancasila," *J. Bina Mulia Huk.*, vol. 5, no. 2, pp. 176–195, 2021.
- [6] R. K. Dewi, Q. J. Ardian, H. Sulistiani, and F. Isnaini, "Dashboard interaktif untuk sistem informasi keuangan pada pondok pesantren mazroatul'ulum," *J. Teknol. Dan Sist. Inf.*, vol. 2, no. 2, pp. 116–121, 2021.
- [7] L. C. Gayatri, "HALO LBH: APLIKASI KONSULTASI HUKUM BERBASIS ANDROID DENGAN FRAMEWORK REACT NATIVE," 2021, *Universitas Pembangunan Nasional Veteran Jakarta*.
- [8] A. W. Sudrajat and I. Inayatullah, "Rancang Bangun Sistem Informasi Konsultasi Hukum Berbasis Android," *J. Teknol. Sist. Inf.*, vol. 2, no. 1, pp. 1–11, 2021.
- [9] S. Riyadi, "Adopsi Metode DevOps Sebagai Acuan Pengembangan Aplikasi Bantuan

- Hukum,” *Gener. J.*, vol. 7, no. 1, pp. 23–30, 2023.
- [10] F. Abdussalaam and M. M. Ramadhan, “Perancangan sistem informasi work order dengan metode iteratif menggunakan framework codeigniter (Studi Kasus: CV Sirna Miskin Bandung),” *J. E-Komtek*, vol. 3, no. 1, pp. 35–48, 2019.