

BAB I

PENDAHULUAN

A. Latar Belakang

Indonesia adalah negara yang memiliki kekayaan budaya yang sangat banyak, seperti setiap daerah memiliki cerita rakyat yang unik dan beragam banyak dipercaya oleh masyarakat setempat bahwa cerita yang terjadi benar dan sering dikaitkan dengan suatu tempat atau nama sebuah tempat, ataupun suatu peristiwa. Cerita rakyat sering kali menjadi salah satu cara untuk menyampaikan nilai-nilai yang mencerminkan kepercayaan, tradisi, dan cara hidup masyarakat setempat. Made Taro juga menegaskan, dalam cerita rakyat terkandung pesan moral serta etika dan nilai-nilai penting tentang kehidupan.

Cerita rakyat yang dari dulu sering kali disampaikan dalam bentuk buku ataupun dari mulut ke mulut yang memiliki nilai-nilai moral yang dapat diterapkan di kehidupan sehari-hari. Tetapi terdapat pengaruh diakibatkan seiring dengan kemajuan teknologi informasi dan komunikasi, khususnya aplikasi web yang kecanggihannya semakin berkembang banyak ditemukan fenomena dimana masyarakat lebih banyak berinteraksi dengan budaya luar. Menurut Agung, M. (2018) Perilaku *imitation* generasi muda saat ini yang lebih mengidolakan budaya luar sehingga menjadi salah satu faktor yang menyebabkan budaya lokal dan nasional ditinggalkan. Hal ini terjadi karena terbatasnya akses dan platform yang

menarik minat untuk mempelajari dan menghargai budaya yang ada dikalangan masyarakat.

Di tengah perkembangan yang ada dimasyarakat saat ini, upaya dalam melestarikan budaya lokal termasuk cerita rakyat namun, banyak cerita rakyat yang mulai terlupakan seperti salah satu cerita rakyat asal usul tari pattuddu yang berasal dari sulawesi barat. Nilai budaya dalam cerita rakyat semakin memudar di kalangan masyarakat karena jarang nya individu yang menceritakan atau membaca cerita rakyat, yang disebabkan oleh kurangnya minat dari pembaca maupun pendengar. Dengan perkembangan teknologi informasi khususnya dalam aplikasi berbasis web yang dapat menjadi inovasi dalam menyampaikan cerita rakyat.

Berdasarkan uraian diatas peneliti merancang sebuah aplikasi cerita rakyat asal usul tari pattudu berbentuk website yang menjadi konsep baru upaya dalam menyebarkan, melestarikan kearifan lokal dan memperkenalkan yang merupakan bagian dari budaya tradisional yang memiliki nilai-nilai yang terdapat didalam cerita rakyat dan dapat meningkatkan minat terhadap warisan budaya. Selain itu aplikasi cerita rakyat dapat disebarkan tidak hanya melalui buku fisik dan disampaikan dari mulut ke mulut saja tetapi dapat dalam bentuk aplikasi yang akan lebih mudah untuk diakses dan diterima oleh semua kalangan.

B. Rumusan Masalah

Berdasarkan latar belakang masalah tersebut dapat dibuat suatu rumusan masalah yaitu:

1. Bagaimana membuat aplikasi yang dapat melestarikan serta memperkenalkan cerita rakyat asal-usul tari pattuddu beserta nilai-nilai budayanya?

C. Tujuan Penelitian

Berdasarkan rumusan masalah, tujuan dari penelitian ini yaitu:

1. Membuat aplikasi web yang menyajikan cerita rakyat dari Sulawesi Barat yang dapat memperkenalkan dan memahami nilai-nilai budaya yang terkandung didalamnya.
2. Menarik minat dalam melestarikan dan mempertahankan warisan budaya.

D. Batasan Masalah

Adapun Batasan masalah dari penelitian ini, berfokus pada perancangan aplikasi cerita rakyat asal usul tari pattudu yang merupakan salah satu cerita rakyat yang berasal dari Sulawesi Barat serta pemahaman nilai-nilai budaya yang terdapat dalam cerita rakyat tersebut.

E. Manfaat Penelitian

Penelitian ini bermanfaat untuk memperkenalkan cerita rakyat asal usul tari pattudu yang merupakan tarian tradisional yang berasal dari Sulawesi Barat, sehingga menjadi sarana dalam melestarikan dan mempertahankan warisan budaya.

BAB II

TINJAUAN PUSTAKA

A. Kajian Teori

Dasar teori yang digunakan pada penelitian ini merupakan pengambilan dari teori-teori yang terkait berupa artikel, jurnal, makalah, tugas akhir dan buku yang ada. Adapun pembahasan teori-teori tersebut sebagai berikut:

1. Cerita Rakyat

Cerita rakyat merupakan cerita yang terdapat dari suatu daerah yang dapat diambil hikmah dan nilai-nilai positif yang bisa diterapkan di kehidupan sehari-hari. Cerita rakyat diceritakan secara turun temurun di masyarakat kepada anak cucunya. Menurut (Ramadhani 2020) cerita rakyat adalah cerita sejak masa lampau yang menjadi ciri khas setiap bangsa yang memiliki kultur budaya yang beraneka ragam mencakup budaya dan sejarah yang dimiliki masing-masing bangsa. Cerita diturunkan dari satu generasi ke generasi lainnya dalam kelompok orang, dari mulut ke mulut.

a. Jenis-jenis Cerita Rakyat

Berikut ini adalah tiga golongan besar dari cerita berdasarkan penelitian William R Bascom, beberapa diantaranya yaitu:

1. Mitos (mite)

Mitos atau mite dapat dipahami sebagai sebuah cerita prosa rakyat yang diyakini kebenarannya setelah tempat dari masyarakat tersebut mengakui

kebenarannya. Mitos ini sendiri diceritakan di dunia lain dan terjadi di masa lampau atau dapat dikatakan bukan di dunia kita saat ini.

2. Legenda

Legenda adalah cerita prosa rakyat yang sebagai sebuah cerita yang dianggap benar-benar terjadi, dimana tempat terjadinya legenda adalah sebuah daerah dan tidak terjadi dalam waktu yang sangat lama, sehingga ada bukti khususnya.

3. Dongeng

Dongeng sendiri memiliki definisi sebagai cerita rakyat berbentuk prosa yang dianggap benar-benar oleh para penyair atau penceritanya. Cerita yang terjadi pada dongen tidak memiliki keterkaitan waktu maupun tempat.

2. Tari Pattuddu dan Sayyang Patuddu

a) Tari Pattuddu



Gambar 2. 1 Tari Pattuddu

Menurut Muhammad Ridwa Alimuddin (2011) Dalam bahasa Mandar, tari diistilahkan “tuqdu”, sedang penari disebut “pattuqduq”. Secara tradisional, tari-tarian merupakan persembahan yang dilakukan kepada dewata. dalam perkembangan selanjutnya, menjadi tarian upacara hiburan pada pesta atau upacara kerajaan. Akhirnya menjadi tarian menghibur untuk umum.

Diperkirakan sejak abad ke-10 tari pattuqduq/tuqduq dikenal oleh masyarakat Mandar. Dikenal berbagai tuqduq, yaitu tuqduq sore, tuqduq sarawadang, tuqduq cakkuriri, tuqduq losa-losa, tuqduq palappa, tuqduq kumba, tuqduq denggo, tuqduq tipalayo, dan tuqduq sawawar. Pattuqduq paling terkenal dan paling dianggap mistis adalah Tuqduq Kumba. Dikenal sebelum I Manyambungi Todilaling menjadi raja Balanipa. Pada zaman dahulu hanya ditampilkan pada upacara kerajaan.

1. Cerita Rakyat Asal Usul Tari Patuddu

Ini adalah kisah tentang seorang pangeran berwajah rupawan dari daerah Mandar, Sulawesi Barat. Ia tinggal di sebuah istana megah. Seluruh halaman istananya dikelilingi pohon rindang dan bunga-bunga indah. Tepat di tengah taman itu, terdapat sebuah kolam pemandian yang terhubung dengan Sungai. Airnya menjadi jernih karena berasal dari balik celah bebatuan.

Pada suatu hari, tepat Ketika hujan gerimis mulai reda, tampaklah pelangi dari atas rumah sang pangeran. Sesungguhnya, itu adalah peristiwa aneh. Apalagi setelah beberapa detik kemudian berhembus aroma harum semerbak. Sang pangeran sangat penasaran. Ia mulai mencari-cari asal bau itu. Ia memasuki setiap ruangan di dalam istana, sampai aroma harum semerbak itu menggiringnya ke taman belakang. Sesampainya di taman, ia sangat terkejut dan kesal. Buah dan bunga-bunganya berantakan seolah ada yang mencurinya dan meninggalkan beberapa di antaranya berserakan di

tanah. “Siapa pun pencurinya, aku akan menangkap dan menghukumnya!” sang pangeran berkata geram.

Ia kemudian membuat sebuah rencana. Pada satu sore, ia sengaja bersembunyi di balik semak untuk mengintai pencuri bunga dan buah di tamannya. Tak lama kemudian, muncullah pelangi warna-warni yang disusul tujuh ekor merpati terbang berputar-putar dengan indah. Pangeran terus mengamati tujuh ekor merpati itu. Lalu, dengan tanpa diduga, tiba-tiba ketujuh ekor merpati itu berubah wujud menjadi bidadari cantik. Rupanya mereka hendak mandi di kolam istana. Sebelum masuk ke dasar kolam, mereka terlebih dahulu bermain-main sambil memetik bunga dan buah sesuka hatinya. Dalam sekejap pangeran justru langsung terpesona melihat kecantikan ketujuh bidadari itu. Ia melupakan niatnya menangkap dan menghukum para pencuri tadi. “Ya Tuhan! Mimpikah aku ini? Cantik sekali gadis-gadis itu,” gumam pangeran dengan nada kagum.

Mereka sedang asyik bermain sambil bersendau-gurau dengan riang. Saat itulah sang pangeran memanfaatkan kesempatan. Dengan hati-hati ia berjalan mengendap-endap dan mengambil selendang milik salah seorang dari ketujuh bidadari itu. Lalu disembunyikannya di balik bebatuan. Setelah puas mandi dan bermain, ketujuh bidadari itu mengenakan selendangnya kembali. Mereka harus segera kembali ke khayangan sebelum pelangi menghilang, karena hanya pelangi satu-satunya jalan kembali ke khayangan. Namun bidadari bungsu tidak menemukan selendangnya. Ia pun

tampak kebingungan. “Selendangku di mana?” Keenam bidadari yang lainnya turut membantu mencari selendang adiknya. Sayangnya selendang itu tetap tidak ditemukan. “Mungkin selendangmu hanyut terbawa arus,” kata bidadari tertua.

Sementara dari kaki langit, pelangi terlihat segera menghilang. Akhirnya keenam bidadari itu meninggalkan si bungsu seorang diri. “Adik, maaf kami harus pergi.” Bidadari bungsu pun menangis sedih, “Ya, Tuhan, bagaimana ini? Aku sebatang kara di bumi ini. Siapa pun nanti yang menolongku, bila ia laki-laki akan kujadikan suamiku dan bila Perempuan akan kujadikan saudari!” Bidadari bungsu memejamkan mata mengucapkan janji.

Tak lama setelah ia berseru demikian, terdengar suara halilintar menggelegar. Seolah lagit telah merestui janji yang diucapkannya. Melihat bidadari bungsu tinggal sendirian, sang pangeran pun keluar dari persembunyian. Ia kemudian datang menghampirinya. “Hai, gadis cantik, kau siapa? Mengapa kau menangis?” tanya pangeran pura-pura tidak tahu. “Aku Kencana, Tuan. Aku tidak bisa pulang ke khayangan karena selendangku hilang,” jawab bidadari bungsu. “Kalau begitu, tinggallah bersamaku,” pinta sang pangeran.

Akhirnya, kencana tinggal dan hidup dengan sang pangeran. Dan, sesuai dengan janji yang diucapkannya, bila seorang lelaki menolongnya, maka ia akan dijadikan suaminya, tak lama setelah itu mereka menikah. Beberapa tahun kemudian, kencana dan sang pangeran dikaruniai seorang

anak laki-laki. Mereka mengasuh anak itu dengan penuh perhatian dan kasih sayang. Terutama kencana, ia hampir melupakan jati dirinya sebagai seorang bidadari. Kehidupan utamanya kini adalah meng-asuh dan mendidik anak. Ia juga menjadi seorang ibu rumah tangga yang patuh pada kodratnya. Setiap hari kencana sangat rajin memasak dan membersihkan rumah.

Lalu suatu hari, Ketika kencana membersihkan kamar dirumah suaminya, tanpa sengaja ia menemukan selendang miliknya yang dulu pernah hilang. Ia sangat terkejut, karena ia tidak pernah menduga jika yang mencuri selendangnya itu adalah suaminya sendiri.

Ia merasa kecewa dengan perbuatan suaminya itu. Hingga ketika suaminya pulang, kencana segera menyerahkan anaknya dan berkata “Suamiku, aku sudah menemukan selendang ku. Maaf, aku harus kembali ke kahyangan untuk menemui keluargaku. Bila kalian merindukanku, pergilah melihat pelangi!” Kencana pun terbang ke angkasa dengan mengipas-ngipas selendangnya. Maka tinggallah sang pangeran bersama anaknya di bumi. Mereka hanya bisa memadangnya berlalu tanpa melontarkan sepatah kata.

2. Gerakan

Menurut Ibu Yuliathy Tahir Guru seni & Pembimbing Sanggar Seni (Wawancara, Laliko: 25 Juni 2024). Tari Patuddu adalah tarian tradisional yang sederhana yang ditarikan oleh perempuan dengan gerakan yang sangat lembut. Tari ini biasa formasinya berjumlah enam ataupun berjumlah genap.

3. Musik Iringan

Musik iringan yang digunakan adalah dua buah Ganrang (gendang) dan satu Ta'bilowe (Gong) yang mengiringi tarian sesuai dengan gerakan penari.

4. Kostum

Kostum yang digunakan dalam tari patuddu merupakan busana khas mandar baju pasangang dan bawahannya dengan menggunakan sarung tenun khas mandar (Lipa' A'di Diratte) Untuk bagian rambut penari biasanya digulung dan diberi hiasan seperti bunga, maupun menggunakan tusuk berwarna emas. Kemudian untuk aksesoris, penari menggunakan gelang, anting dan kalung khas Mandar.

5. Properti

Tari patuddu propertinya yang digunakan dalam tarian ini yaitu menggunakan kipas dan selendang.

b) Sayyang Pattudu



Gambar 2. 2 Sayyang Pattuddu

Menurut Muhammad Ridwan Alimuddin (2020) Tradisi sayyang pattuqdu di Mandar tidak diketahui persis kapan mulai dilakukan. Diperkirakan tradisi itu dimulai ketika Islam menjadi agama resmi beberapa kerajaan di Mandar, kira-kira abad XVI. Sayyang pattuqdu awalnya hanya berkembang di kalangan istana, yang

dilaksanakan pada perayaan Maulid Nabi Muhammad SAW. Kuda digunakan sebagai sarana sebab dulunya di Mandar, kuda adalah alat transportasi utama.

Dalam perkembangannya, sayyang pattuqdu menjadi alat motivasi bagi anak kecil agar segera menamatkan Al-Quran. Ketika seorang anak kecil mulai belajar Al-Quran, oleh orang tuanya dijanji akan diarak keliling kampung dengan sayyang pattuqdu jika khatam Al-Quran. Karena ingin segera naik kuda penari, maka sang anak ingin segera pintar mengaji dan khatam Al-Quran. Lewat buku “Saiyyang Pattuqduq dan Khataman Al Quran di Mandar”, Ma’lum Rasyid menyimpulkan tentang kemunculan tradisi “saeyyang pattuqduq totammaq mangaji”. Bahwa itu diawali oleh Daetta Tommuane untuk kemudian dikembangkan oleh Sayyid Alwi bin Abdullah bin Sahl Jamalullael.

1) Cerita Rakyat Asal Usul Sayyang Pattudu

Daetta Tommuane yang pertama memiliki saiyyang pattuqdu,

Daeta Tommuane juga yang pertama kali menunggangi saiyyang pattuqdu, dan

Daeta Tommuane juga yang pertama kali melantunkan kalindaqdaq.

Lebih lanjut beliau menceritakan:

Ada pengawal Raja yang khusus memelihara kuda sang Raja, nama kuda ini Tembaga Parepulu melapor kepada raja “saya tidak tahu tuan sudah tiga hari ini saya perhatikan kalau kuda itu saya datang mandikan selalu menari di kandang” lalu Raja berkata ”bernarkah itu?” “begitulah tuan” tutur pengawal. Akhirnya Raja berkata “besok sore pergi mandikan itu kuda saya mau melihat dan saya menunggu kamu di atas sondoq teras rumah panggung), jika kamu pergi mandikan hendaknya kuda itu berlari begitupun pada saat

Kembali” Keesokan harinya pengawal rajapun pergi memandikan kuda tersebut kemudian raja menunggu diatas sondoq.

Pada saat pengawal datang memandikan ternyata raja sudah tidak di atas sondoq beliau masuk ke dalam rumah berbaring sampai tertidur, lalu pengawal ini bergumam “dimanakah sang raja, mengapa tidak ada di atas sondoq?” Akhirnya pengawal ini naik ke rumah ternyata Raja ada dibagian tengah rumah tertidur, lalu pengawal ini membangunkan Raja dan menyampaikan bahwa “kuda itu menari lagi tuan”. Raja pun turun melihat ternyata benar kuda itu menari, apabila kandangnya dipukul-pukul kuda itu semakin menari. Lalu Raja berkata “mungkin kuda ini mau ditunggangi”, akhirnya Raja menungganginya dan kuda itupun terus mattuqdu (menari). Sewaktu Raja di atas kuda istrinya melihat turun dan berkeinginan untuk ikut menunggangi, akhirnya Raja turun dan diantarlah kuda itu ke tangga untuk menjemput sang Permaisuri Daeta Tommuane, kemudian Raja memerintahkan terhadap dua pengawal untuk menjaga Permaisuri. Pada saat anaknya perempuan mengintip turun anak inipun meminta untuk ikut menungganginya akhirnya anaknya pun dinaikkan, kemudian diperintahkan lagi pengawal raja dua orang dan anaknya menjadi empat orang, masing-masing dua di sebelah kanan dan dua di sebelah kiri dan inilah yang disebut dengan pesarung yang sampai sekarang jumlahnya empat orang. Kuda ini terus menari (sementara permaisuri dan sang putri berada di atas kuda) maka dengan spontan Mara'dia melantunkan kalindaqdaq.

Permisi Permaisuriku yang mulia Permisi Intan Jamrudku Dikaulah pohon lambeq dan baraneq Di Tanah Balanipa Tempat berlindungnya Lelupang

sumaqhuri Jika bergoyang Pohon lambeq dan baraneq Resahlah Lelupang Sumaqhuri Jika engkau ingin harum Di tanah Balanipa Siram-siramlah Lelupang Sumaqhuri. Sesudah kejadian itu secara spontan Raja berkata kepada putrinya “berlajarlah mengaji nak kalua engkau tamma mengaji” saya akan naikkan kamu ke atas kuda pattudu dan saya akan membawa kamu keliling kampung, setelah anak khatam alquran maka Rajapun memenuhi janjinya.

2) Gerakan Kuda

Ada dua gerakan utama dalam tarian kuda, yaitu gerakan kepala yang mendongak-dongak, dan gerakan dua kaki ke depan yang dihentakkan secara bergantian ke tanah.

3) Musik Iringan

Parrabawa orang yang memainkan (Rebana). Dan diselengi dengan kalindaqdaq (syair puisi khas mandar yang dilantunkan).

4) Kostum

Baju pasangang yang digunakan penunggang kuda atau biasa juga disebut “boko/pokko”. Adapun sarung yang dipadukan mengenakan sutera corak (sureq) “salaka” yang berwarna hitam dengan kotak-kotak berwarna perak. gadis penunggangnya yang disebut dengan *Pesaweang* dan dinaungi dengan payung kehormatan yang pembawa payung disebut *Palla'lang*. Untuk penamatan Qur'an, dalam bahasa setempat disebut *mappatamma* Qur'an.

5) Properti

Untuk perhiasan lainnya secara tradisional yaitu, di kepala mengenakan “simbolong tibaliq” yang berhiaskan Melati, anting putih yang berbalut kapas

berupa “dali”, di leher hanya seuntai kalung dengan “ringgiq bulabang” (ringgit emas), beberapa gelang “buwur” dan kipas di tangan.

6) Pelaksanaan Sayyang Pattudu

Secara umum, ada lima bentuk pelaksanaan “saeyyang pattuqduq”. digunakan di acara Maulid Nabi Muhammad SAW, Kedua “saeyyang pattuqduq” digunakan sebagai pemenuhan nazar, ketiga “Saeyyang pattuqduq” juga digunakan sebagai bagian acara dalam penyambutan tamu-tamu penting. Keempat, “saeyyang pattuqduq” sebagai sebuah acara festival. Terakhir, ketika “saeyyang pattuqduq” menjadi rangkaian acara lain. Misal saat “saeyyang pattuqduq” menjadi tema utama penampilan tim Sulawesi Barat di Pawai Budaya Nusantara.

3. *Website*

Website adalah kumpulan halaman web yang saling terhubung dan dapat diakses melalui Internet. Halaman-halaman ini biasanya berisi informasi dalam bentuk teks, gambar, video dan multimedia lainnya dan diidentifikasi dengan nama domain atau alamat *Uniform Resource Locator (URL)*. Sebuah situs web dapat dibuat dan dioperasikan oleh individu, organisasi, perusahaan atau pemerintah untuk berbagai tujuan, termasuk tujuan informasi, hiburan, pendidikan dan komersial. Berikut beberapa jenis – jenis kategori website:

a) *Website Statis*

Website Statis merupakan web yang isi atau kontennya tidak pernah di perbaharui atau di *update* dalam durasi waktu tertentu, atau bisa di katakan isinya

tidak pernah berubah sama sekali, contoh website statis seperti *web company profile* suatu instansi atau perusahaan dll.

b) *Website Dinamis*

Situs web yang dinamis adalah situs web yang konten atau elemen lainnya sering berubah karena pengguna atau sering diupdate sesering mungkin. *Website dinamis* ini biasanya cocok di gunakan untuk web blog, jejaring sosial, toko online dll.

c) *Web Interaktif*

Merupakan website yang berinteraksi antara penggunanya. Biasanya berupa forum diskusi maupun blog. Dimana adanya pengguna dapat mengedit di situs web interaktif.

4. Animasi 3D

Animasi berasal dari kata "*Animation*" yang dalam bahasa Inggris "*to animate*" yang berarti menggerakkan. Jadi animasi dapat diartikan sebagai menggerakkan sesuatu (gambar atau objek) yang diam.

Animasi 3D adalah teknik untuk menciptakan gambar bergerak dalam ruang tiga dimensi dengan menggunakan perangkat lunak komputer. Teknik ini memungkinkan penciptaan objek dan karakter yang dapat bergerak dan berinteraksi dalam lingkungan virtual. Animasi 3D digunakan secara luas dalam berbagai industri, termasuk film, video game, simulasi medis, arsitektur, dan pendidikan.

1. Modelling Proses ini adalah proses pembuatan model objek dalam bentuk 3D dikomputer. Model bisa berupa karakter (makhluk hidup), seperti manusia,

hewan, atau tumbuh-tumbuhan, atau berupa benda mati seperti rumah, mobil, peralatan, dan lain-lain.

2. *Texturing* Proses ini adalah proses pembuatan dan pemberian warna dan material (*texture*) pada objek yang dimodelkan sebelumnya sehingga akan tampak kesan yang nyata. Material atau *texture* dapat berupa foto atau gambar yang dibuat dengan aplikasi *software* 3D atau dengan bantuan software digital seperti Photoshop, dan lain-lainnya.
3. *Lighting* adalah proses pembuatan dan pemberian cahaya pada model sehingga terdapat kesan visual yang realistis, karena terdapat kesan, kedalaman, ruang dan pembayangan objek. Memberikan fitur global *illumination*, yang mampu memberikan hasil pencahayaan yang realistis dan natural, seperti dalam kondisi nyata.
4. *Environment Effect* Proses ini adalah proses pembuatan panorama lingkungan pada objek model yang akan semakin menambah kesan realistis. *Environment* mencakup background pemandangan atau langit, dan lainlain. Juga mencakup pembuatan efek-efek 3D yang diperlukan, seperti efek api, air, asap, kabut, dan efek-efek lainnya.
5. *Animation* adalah proses pembuatan animasi untuk model. Animasi dapat berupa gerakan, yang dapat menentukan arah dimulai suatu gerakan animasi, disesuaikan dengan *storyboard* yang telah dibuat.
6. *Rendering* adalah proses pengkalkulasian pada model 3D yang telah diberi *texture*, *lighting*, *environment effect*, dan *animation*. Dimana hasil animasi yang didapatkan tampak sangat nyata dan menarik.

5. *Visual Studio Code*



Gambar 2. 3 Logo *Visual Studio Code*

Visual Studio Code adalah editor kode sumber yang dikembangkan oleh Microsoft yang mudah digunakan di desktop untuk sistem operasi Windows, macOS, dan Linux. Memiliki ekosistem ekstensi yang luas untuk bahasa dan *runtime* lain seperti C++, C#, Java, Python, PHP, Go, dan.NET, serta dukungan bawaan untuk JavaScript, TypeScript, dan Node.js.

6. **Blender**



Gambar 2. 4 Logo *Blender*

Blender merupakan perangkat lunak komputer 3D yang bersifat *open source*. Perangkat lunak ini bisa digunakan untuk membuat animasi dan simulasi. Menurut Roosendal (2022), Aplikasi Blender adalah aplikasi *open source* memiliki fitur lengkap dalam membuat sebuah animasi baik itu 3D maupun 2D dalam Blender terdapat istilah *vertices*, *edge* dan *face*. *Vertices* merupakan objek berupa titik. *Edge* merupakan garis yang terbentuk dari dua *vertice*.¹⁷ Sedangkan *face* merupakan bidang yang terbentuk minimal dari tiga *vertice* yang saling terhubung. Manipulasi ketiga dasar itulah yang membentuk sebuah objek tiga dimensi baru yang yang diinginkan. Berikut adalah gambar tampilan awal aplikasi blender.



Gambar 2. 26 *Interface Blender*

Berikut beberapa fitur yang umum digunakan dalam aplikasi blender untuk membuat animasi 3D:

1. Transformasi (*Transformation*)

Trasnformasi adalah salah satu dasar dalam blender yang digunakan untuk mengontrol *position*, *rotasi*, dan *scale* objek dalam ruang 3D.

2. *Keyframes*

Keyrrames adaalah elemen yang digunakan dalam animasi untuk transformasi yang relevan posisi, rotasi, dan penskalaan objek/karakter tertentu. Menjalankan animasi, blender akan *menginterpolasi* parameter transformasi yang ditentukan antara *Keyframes*, sehingga mendapatkan gerakan yang halus.

3. *Texturing* dan *Shading*

Texturing dan *shading* adalah bagian penting dalam blender dari proses rendering, yang menentukan bagaimana objek terlihat dengan material tertentu.

4. *Rigging*

Rigging merupakan fitur yang sangat penting dalam proses penganimasian karakter dalam Blender. Ini memungkinkan pengaturan anggota karakter dengan cara yang jauh lebih efisien daripada harus secara manual menyesuaikan

5. *Lighting*

Pencahayaan (*lighting*) adalah faktor penting dalam membuat adegan terlihat realistis.

6. *Rendering*

Rendering adalah proses akhir di mana adegan 3D yang telah diatur (model, tekstur, pencahayaan, dan animasi) diubah menjadi gambar atau video.

7. **PHP (*Hypertext Preprocessor*)**

PHP atau Hypertext Preprocessor adalah sebuah bahasa script berbasis server (server-side) yang mampu mem-parsing kode php dari kode web dengan ekstensi .php, sehingga menghasilkan tampilan website yang dinamis di sisi client (browser). Dengan menambahkan skrip PHP, anda bisa menjadikan halaman HTML menjadi lebih powerful, dinamis dan bisa dipakai sebagai aplikasi lengkap, misalnya web portal, e-learning, e-library, dll.

8. **MySQL (*My Structured Query Language*)**

MySQL adalah sistem manajemen basis data relasional (RDBMS) sumber terbuka. Pada tahun 2013, MySQL merupakan RDBMS kedua yang paling banyak digunakan di dunia dan yang pertama untuk RDBMS sumber terbuka. MySQL didasarkan pada SQL (Select Query language) yaitu bahasa standar untuk berkomunikasi dengan sistem manajemen basis data relasional.

MySQL menggunakan bahasa pemrograman C dan C++ serta dapat berjalan di semua platform seperti Windows, Linux dan Unix. MySQL merupakan database pilihan untuk beberapa bahasa pemrograman web seperti PHP, Ruby on Rails dan

Python. Jadi MySQL adalah database management system yang menggunakan bahasa SQL sebagai bahasa penghubung antara perangkat lunak aplikasi dengan database server. MySQL mempunyai beberapa kelebihan yang bisa dimanfaatkan untuk mengembangkan perangkat lunak yang ada seperti:

- a) Mendukung Integrasi dengan bahasa pemrograman lain. Website atau perangkat lunak terkadang dikembangkan dengan menggunakan berbagai macam bahasa pemrograman, jadi tidak perlu khawatir jika menggunakan MySQL. Maka dari itu, MySQL bisa membantu untuk mengembangkan perangkat lunak yang lebih efektif dan tentu saja lebih mudah dengan integrasi antara bahasa pemrograman.
- b) Tidak membutuhkan RAM besar. MySQL dapat dipasang pada server dengan spesifikasi kecil. Jadi tidak perlu khawatir jika hanya mempunyai server dengan kapasitas 1 GB karena masih bisa menggunakan MySQL sebagai database.
- c) Mendukung multi user. MySQL dapat dipakai oleh beberapa user dalam waktu bersamaan tanpa membuatnya crash atau berhenti bekerja. Ini dapat dimanfaatkan ketika mengerjakan proyek yang sifatnya tim sehingga seluruh tim dapat bekerja dalam waktu bersamaan tanpa harus menunggu user lain selesai.

9. XAMPP

Xampp adalah software, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (localhost), yang terdiri atas program Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP

dan Perl. Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU General Public License dan bebas, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis. Untuk mendapatkannya dapat mendownload langsung dari web resminya.

10. Bootstrap

Bootstrap merupakan framework HTML, CSS, Javascript populer untuk membangun situs web yang *responsive*. Dengan Bootstrap, developer website dapat membuat responsive website dan dapat berjalan sempurna pada browser-browser populer seperti *Chrome*, *Firefox*, *Opera*, dan *Internet Explorer* serta pada perangkat-perangkat *handphone* maupun PC desktop. *Responsive Web Design* adalah desain website yang secara otomatis akan menyesuaikan dengan kondisi platform yang digunakan dari sisi tampilan serta segala isinya yang ada pada website tersebut.

Bootstrap terdiri dari satu set file CSS, *Javascript*, dan *Jquery*. Bootstrap menyediakan *class-class* dalam membuat berbagai jenis komponen-komponen website seperti *menu buttons*, *card*, *form*, *jumbotron*, *media object*, *navbar*, *pagination*, *progress*, *spinners*, dan lainnya.

11. Javascript

Javascript adalah satu-satunya bahasa web yang dapat membuat beranda situs web Anda lebih menarik dan dinamis. Javascript merupakan suatu bahasa script yang banyak digunakan dalam dunia teknologi terutama internet, bahasa ini

dapat bekerja di sebagian besar web browser seperti *Internet Explorer (IE)*, *Mozilla Firefox*, *Netscape*, *opera* dan web browser lainnya.

12. HTML (HyperText Markup Language)

Hypertext Markup Language atau biasa disebut HTML adalah bahasa standar pemrograman untuk membuat suatu website yang bisa diakses dengan internet. Dengan kata lain halaman website yang dilihat dan dibaca disusun dengan menggunakan bahasa ini dan kemudian diterjemahkan oleh komputer agar dapat dipahami oleh penggunanya.

Istilah markup pada html sendiri diartikan sebagai suatu hal yang dilakukan tag html terhadap teks yang berada didalamnya, misal jika mengetik suatu teks dengan tanda tag maka teks tersebut akan muncul sebagai teks dengan huruf tebal atau bold pada suatu laman website. Simbol atau tag html tersebut ditulis pada laman html yang sudah disediakan pada dashboard website pada umumnya. Sementara language dalam html adalah bahasa pemrograman atau script yang disusun dari tag-tag tertentu yang nantinya akan diterjemahkan kedalam teks atau visual yang dapat dilihat pada website.

HTML yang dikenal sebagai bahasa pemrograman dalam ilmu komputer memiliki beberapa fungsi, diantaranya adalah sebagai berikut (Alfarabi Hadi, n.d.):

- a) Fungsi utama html yang diketahui adalah untuk membuat suatu halaman website yang bisa dibaca dan dipahami oleh pengguna dengan lebih mudah. Seluruh laman website yang ada dalam internet dibuat dengan html dan tidak ada pengecualian.

- b) Menandai teks pada suatu laman, html ditulis pada suatu halaman dokumen dengan tag atau simbol tertentu dimana simbol dan tag tersebut akan menandai teks menjadi tebal, miring, bergaris tebal dan lain sebagainya. Misal jika membuat suatu teks menjadi teks miring atau italic, dalam laman html dituliskan kode < i >, < b > untuk teks tebal dan < u > untuk teks bergaris bawah.

13. CSS (*Cascading Style Sheets*)

CSS = *Cascading Style Sheets*. CSS merupakan bahasa yang digunakan untuk mengatur tampilan suatu dokumen yang ditulis dalam bahasa *markup / markup language*. apabila kita berbicara dalam konteks web, bisa diartikan secara bebas sebagai: CSS merupakan bahasa yang digunakan untuk mengatur tampilan / desain suatu halaman HTML.

CSS dapat mengubah font, ukuran font, warna dan format font. mengatur ukuran layout, lebar, tinggi dan warna element, mengubah tampilan form, membuat halaman website yang responsive dan masih banyak lagi. Untuk mendesain font dapat dilakukan dengan mendefinisikan font, untuk mengatur warna bisa menggunakan color, margins digunakan untuk mengatur jarak pada luar element tertentu. mengatur warna atau gambar pada latar belakang bisa menggunakan “background”. mengatur ukuran font gunakan “font size”. jenis font menggunakan “font-family” dan banyak lagi lainnya.

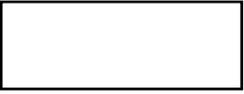
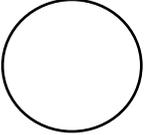
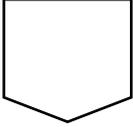
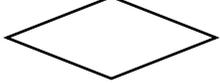
14. *Flowchart*

Menurut (Ayu, 2021) *Flowchart* adalah bentuk Diagram yang digunakan untuk menggambarkan alur, proses, atau sistem suatu data. Biasanya, *flowchart* ini

sering digunakan untuk menyajikan data nama kepengurusan dalam suatu organisasi maupun perusahaan. Bukan hanya itu, *flowchart* juga biasa digunakan dalam dunia bisnis untuk menyusun rencana pemasaran atau produksi yang akan dilakukan. Dalam hal ini, *flowchart* selalu mempunyai bagian permulaan dan akhir yang membantu pembaca memahami alur informasi yang ada.

Berikut ini bagian *flowchart* yang dibuat dengan simbol-simbol:

Tabel 2. 1 *Symbol Flowchart*

Simbol	Keterangan
	Simbol <i>input/output</i> digunakan untuk mewakili data <i>input/output</i> .
	Simbol proses, digunakan untuk mewakili suatu proses.
	Simbol proses terdefinisi, digunakan untuk menunjukkan suatu operasi yang rincinya ditunjukkan di tempat lane.
	Simbol titik terminal, digunakan untuk awal dan akhir proses.
	Simbol penghubung, digunakan untuk menunjukkan sambungan dari alir yang terputus di halaman yang masih sama.
	Simbol penghubung, digunakan untuk menunjukkan sambungan dari alir yang terputus di halaman yang berbeda
	Simbol garis alir, menunjukkan arus dari proses.
	Simbol Keputusan, digunakan untuk suatu penyelesaian kondisi di dalam program.

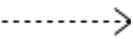
15. *Unified Model Language (UML)*

Unified Model Language (UML) adalah bahasa untuk menspesifikasi, memvisualisasikan, membangun dan mendokumentasikan *artifacts* (bagian dari informasi yang digunakan untuk dihasilkan oleh proses pembuatan perangkat lunak, *artifact* tersebut dapat berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya.

Selain itu, UML adalah Bahasa pemodelan yang menggunakan konsep orientasi object. UML dibuat oleh Grady Booch, James Rumbaugh, dan Ivar Jacobson di bawah bendera Rational Software Corps. UML menyediakan notasi-notasi yang membantu memodelkan sistem dari berbagai perspektif. UML tidak hanya digunakan dalam pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan.

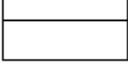
Adapun daftar simbol-simbol UML yaitu:

Tabel 2. 2 *Symbol Use Case Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
3		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).

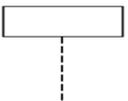
NO	GAMBAR	NAMA	KETERANGAN
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

Tabel 2. 3 *Symbol Class Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.

NO	GAMBAR	NAMA	KETERANGAN
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.

Tabel 2. 4 *Symbol Sequence Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

Tabel 2. 5 *Symbol StateChart Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>State</i>	Nilai atribut dan nilai link pada suatu waktu tertentu, yang dimiliki oleh suatu objek.
2		<i>Initial Pseudo State</i>	Bagaimana objek dibentuk atau diawali

3		<i>Final State</i>	Bagaimana objek dibentuk dan dihancurkan
4		<i>Transition</i>	Sebuah kejadian yang memicu sebuah state objek dengan cara memperbaharui satu atau lebih nilai atributnya
5		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
6		<i>Node</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

Tabel 2. 6 *Symbol Activity Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

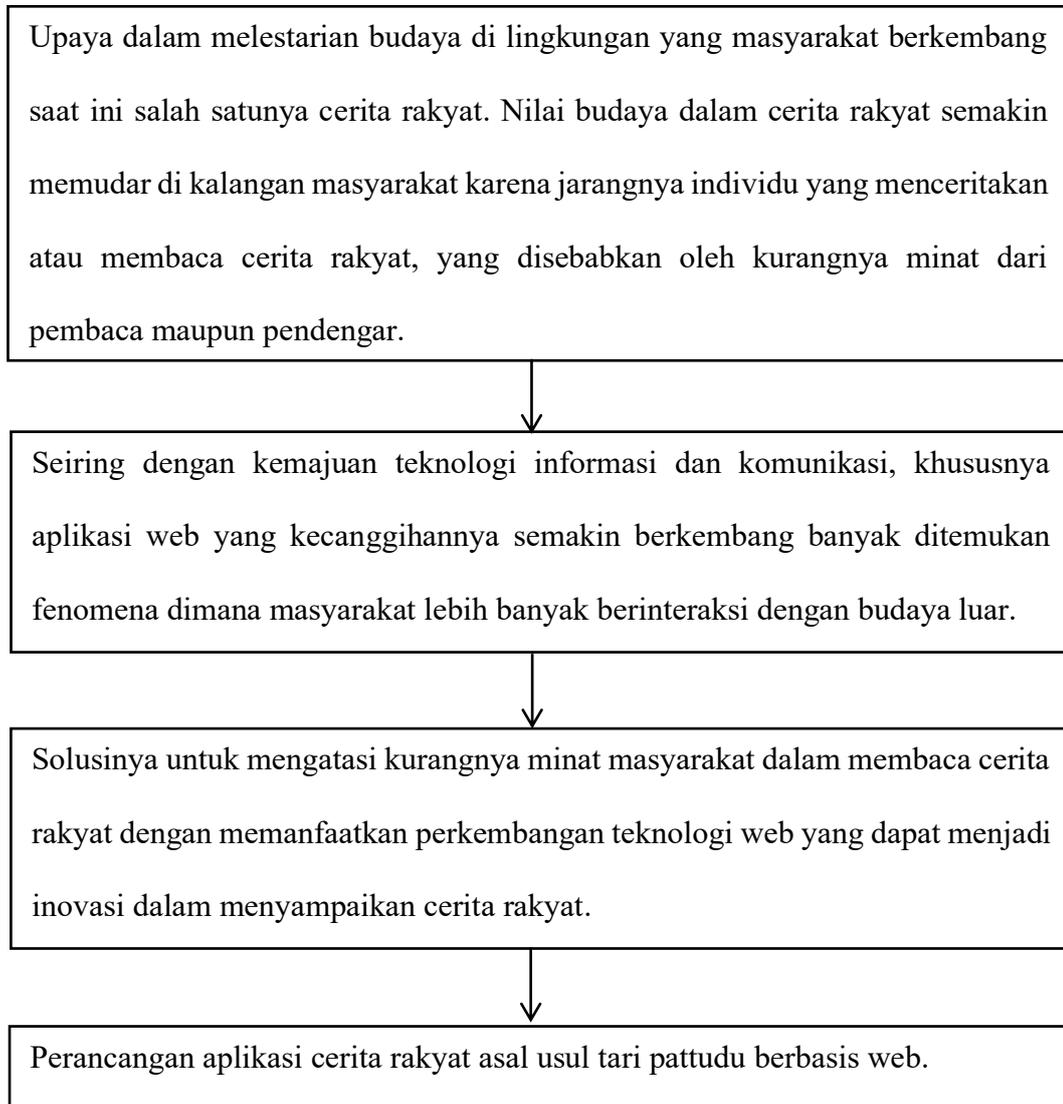
B. Kajian Hasil Penelitian Terdahulu

Berdasarkan penelitian terdahulu menjadi salah satu acuan penulisan dalam melakukan penelitian sehingga penulis dapat memperkaya teori yang digunakan dalam mengkaji penelitian yang dilakukan. Dari penelitian terdahulu dengan judul yang sama seperti referensi dalam memperkaya bahasa kajian pada penelitian

penulis, berikut merupakan penelitian terdahulu berupa beberapa jurnal terkait penelitian yang dilakukan penulis.

1. Penelitian sebelumnya mengenai cerita rakyat juga yang dilakukan oleh (Feby Febyanti, dkk. 2023) Perancangan Web Komik Angling Dharma Sebagai Pengenalan Cerita Rakyat Jawa Timur Kepada Desawa 18-25 Tahun dalam hal ini hasil penelitian Cerita Rakyat Angling Dharma memiliki potensi untuk dikenal oleh generasi muda menjadi media sangat penting untuk menyampaikan dan mengenalkan cerita rakyat kepada target audiens, salah satu media yang efektif adalah web.
2. Penelitian sebelumnya mengenai cerita rakyat juga yang dilakukan oleh (Reggy Cahyari, dkk, 2020) dengan judul Ilustrasi desain dalam media interaktif cerita rakyat sukabumi berbasis aplikasi untuk anak sekolah dasar di sd islam at tarbiyah dengan hasil penelitian pentingnya membuat terobosan baru dengan memanfaatkan teknologi seperti media interaktif untuk melestarikan atau mengenalkan Kembali cerita rakyat yang berasal dari daerah sukabumi.
3. Penelitian sebelumnya mengenai cerita rakyat juga yang dilakukan oleh (Syamsul Bakhri, 2023) dengan judul perancangan animasi interaktif cerita rakyat asal usul nama kota jambi berbasis android dengan hasil bahwa aplikasi perancangan animasi interaktif cerita rakyat ini mampu digunakan untuk membantu anak-anak belajar mengetahui tentang kebudayaan. Selain itu penggunaan aplikasi tersebut juga cukup mudah dimengerti oleh pengguna anak-anak.

C. Kerangka Pikir



BAB III

METODE PENELITIAN

A. Jenis Penelitian

Jenis penelitian yang digunakan dalam ini menggunakan pendekatan kualitatif yang mengutamakan fokus pada pemahaman mendalam tentang budaya dan cerita rakyat, Penelitian dengan pendekatan kualitatif memungkinkan untuk menggali makna, nilai, dan pengalaman yang mendalam.

B. Lokasi dan Waktu

Penelitian dilakukan di dua lokasi, yaitu terletak di Desa Laliko, Kecamatan Campalagian, Kabupaten Polewali Mandar, Sulawesi Barat, dan Pinrang, Sulawesi Selatan. Penelitian ini akan berlangsung selama \pm 2 bulan tahun 2024.

C. Alat dan Bahan

Untuk melakukan proses penelitian, maka yang harus diperlukan adalah alat dan bahan penelitian, guna mendukung kegiatan tersebut. Adapun alat dan bahan sebagai berikut:

1. Alat yang digunakan selama proses penelitian yaitu:
 - a. laptop Acer TravelMate P449 dengan spesifikasi:
 - 1) *Processor*: Intel® Core i5
 - 2) *RAM*: 8 GB
 - 3) *SSD*: 256 GB
 - 4) *LCD Monitor*: 14"

2. Perangkat lunak (*Software*)

Perangkat lunak yang digunakan dalam perancangan animasi yaitu:

- 1) *Sistem Operasi Windows 10*
- 2) *Visual Studio Code*
- 3) *Blender*

D. Metode Pengumpulan Data

Adapun langkah-langkah yang penelitian tempuh untuk pengumpulan data ialah sebagai berikut:

1. Wawancara

Wawancara yang dilakukan dengan memberikan pertanyaan kepada sumber yang dapat dipercaya secara langsung guna mendapatkan data yang benar dan akurat.

2. Dokumentasi

Dalam penelitian ini, penelitian melakukan teknik pengumpulan data dengan cara pendokumentasian dilapangan serta mencari dokumen-dokumen terkait sayyang pattuddu dan tari pattudu, seperti catatan Sejarah, buku, rekaman video, dan sumber-sumber lainnya.

3. Studi *Literatur*

Pengumpulan data dengan cara mengumpulkan data-data yang diperoleh dari buku, media maupun dari hasil penelitian orang lain untuk menyusun dasar teori yang digunakan dalam melakukan penelitian ini.

E. Tahap Penelitian

Tahapan penelitian yang dimaksud dalam penelitian ini ada beberapa tahapan yaitu persiapan penelitian, pengumpulan data analisis, perancangan,

pengujian dan implementasi. Adapun Uraian dari tahapan tersebut adalah sebagai berikut:

1. Persiapan Penelitian

Pada tahapan ini peneliti melakukan persiapan penelitian. Persiapan penelitian yang dimaksud adalah menyiapkan buku-buku, artikel-artikel tentang topik penelitian serta *software* yang digunakan selama penelitian.

2. Studi *Literature*

Pada tahapan ini peneliti melakukan apa yang disebut dengan kajian pustaka, yaitu mempelajari buku-buku referensi dan hasil penelitian sejenis sebelumnya yang pernah dilakukan oleh orang lain. Tujuannya ialah untuk mendapatkan landasan teori mengenai masalah yang akan diteliti.

3. Pengumpulan Data

Pada tahap ini peneliti mencari referensi sebanyak-banyaknya, serta melakukan pencatatan dan pengamatan secara tidak langsung.

4. Analisis

Pada tahap analisis, peneliti melakukan analisa terhadap sistem yang di terapkan sekarang berdasarkan kemudian merumuskan masalah yang menjadi pokok penelitian sehingga dapat dibuat alternatif pemecahan masalah.

5. Perancangan Peneliti

kemudian merancang aplikasi yang ingin dibuat berdasarkan alternatif pemecahan masalah.

6. Pengujian

Setelah melakukan perancangan, peneliti kemudian menguji hasil perancangan yang telah dibuat. Jika hasil perancangan terdapat kekurangan atau kelemahan maka kembali ke tahap analisis.

7. Implementasi

Setelah pada perancangan tidak terdapat kekurangan maka animasi siap untuk menyaksikan oleh user.

F. Metode Pengujian

Pengujian sistem dilakukan untuk memastikan bahwa aplikasi berjalan sesuai dengan yang diharapkan. Pengujian dilakukan dengan dua metode utama yaitu *Black Box Testing* dan *White Box Testing*:

1. *Black Box Testing*

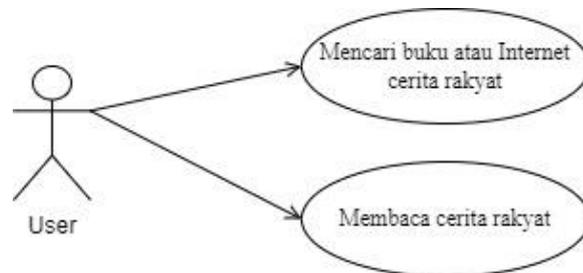
Black Box Testing adalah metode pengujian perangkat lunak yang fokus pada pengujian fungsionalitas aplikasi tanpa melihat ke dalam struktur kode atau implementasi internal. Dalam *Black Box Testing*, penguji hanya berinteraksi dengan antarmuka pengguna dan memeriksa apakah aplikasi memberikan output yang diharapkan berdasarkan input tertentu.

2. *White Box Testing*

White Box Testing adalah metode pengujian perangkat lunak yang fokus pada pengujian logika internal dan struktur kode aplikasi. Dalam *White Box Testing*, penguji harus memiliki pemahaman yang mendalam tentang struktur kode, alur logika, dan jalur eksekusi yang ada dalam aplikasi.

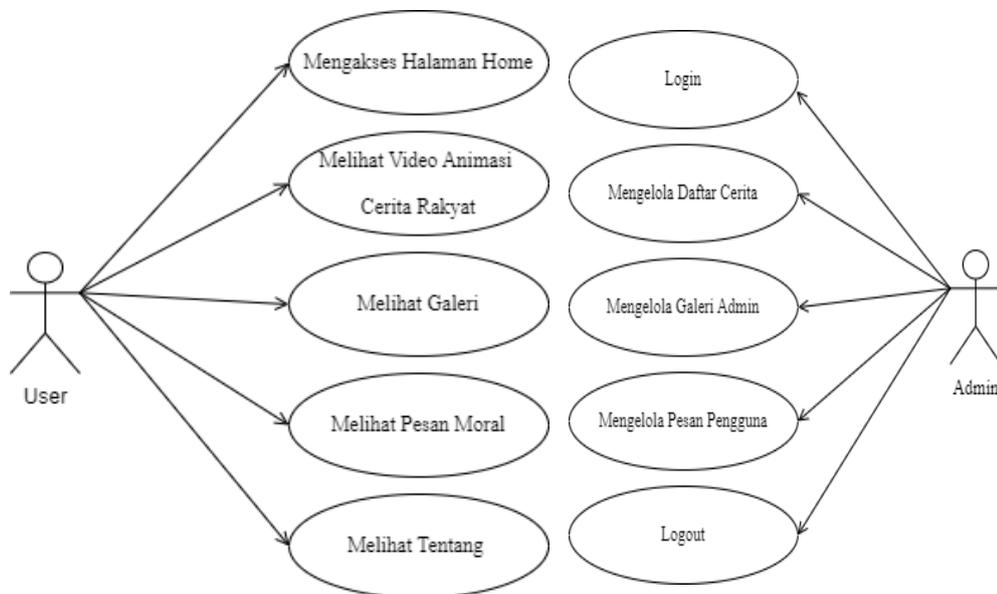
G. Desain Sistem

1. Desain sistem yang berjalan



Gambar 3. 3 Desain Sistem yang Berjalan

2. Desain sistem yang diusulkan



Gambar 3. 4 Desain Sistem yang Diusulkan

BAB IV

HASIL DAN PEMBAHASAN

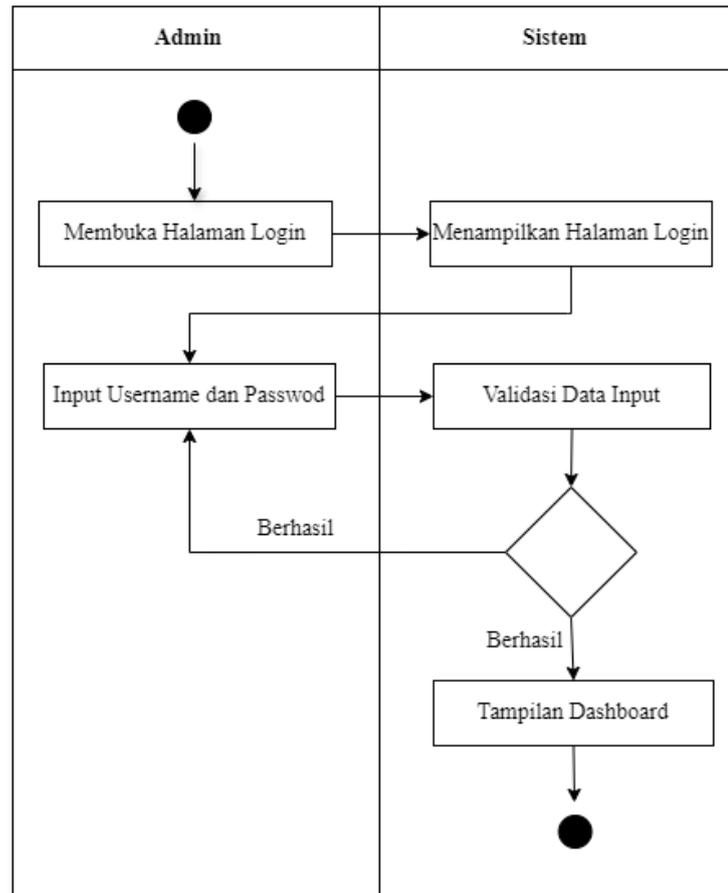
A. Analisis Aliran Data UML

Adapun tujuan dalam perencanaan sistem ini adalah untuk menyebarkan, melestarikan dan menggali lebih dalam tentang salah satu budaya dan tradisi cerita rakyat yang berasal dari Sulawesi barat. Pada kesempatan ini, peneliti merancang aplikasi cerita rakyat yang berorientasi objek sehingga menggunakan berbagai Diagram seperti *Activity Diagram* dan *sequence Diagram*.

1. *Activity Diagram*

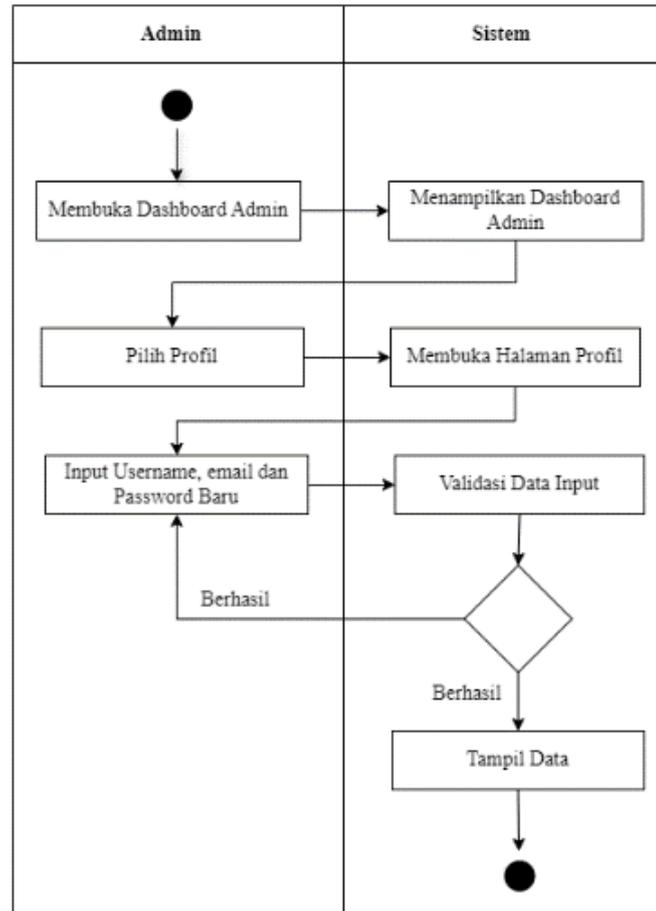
Activity Diagram adalah bahasa spesifikasi standar yang digunakan untuk mendokumentasikan, menspesifikasikan, dan membangun perangkat lunak. Diagram ini menggambarkan berbagai aktivitas yang terjadi dalam alur proses suatu sistem, memberikan pandangan yang jelas tentang bagaimana proses tersebut berjalan dari awal hingga akhir. Dengan menggunakan *activity Diagram*, pengembang dapat mengidentifikasi dan memvisualisasikan setiap langkah dalam proses, sehingga memudahkan dalam perancangan dan implementasi perangkat lunak yang sesuai dengan kebutuhan sistem.

a. *Activity Diagram Admin*

1) *Activity Diagram Login Admin*

Gambar 4. 1 *Activity Diagram Login Admin*

Pada gambar diatas menjelaskan proses login dari admin. Pertama admin membuka halaman login, kemudian sistem menampilkan halaman login. Selanjutnya, admin memasukkan username dan password, yang diterima oleh sistem. Sistem memeriksa username dan password jika berhasil, sistem akan menampilkan dashboard admin. Jika gagal, admin akan kembali ke login untuk menginput kembali username dan password.

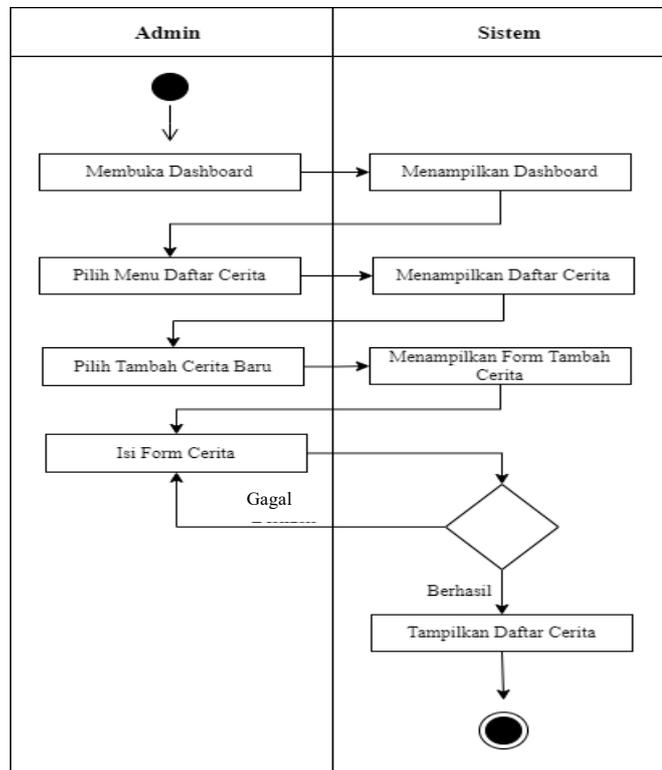
2) *Activity Diagram* Profil Admin

Gambar 4. 2 *Activity Diagram* Profil Admin

Pada gambar diatas menjelaskan proses login dari profil admin. Pertama admin membuka halaman *dashboard* admin, kemudian sistem menampilkan halaman *dashboard* admin. Selanjutnya, admin pilih menu profil dan sistem menampilkan halaman profil. Pada halaman profil admin dapat mengubah *Username*, *Email*, dan *Password* baru. Sistem memeriksa username dan password jika berhasil, sistem akan menampilkan dashboard admin. Jika gagal, admin akan menginput kembali *Username*, *Email*, dan *password*.

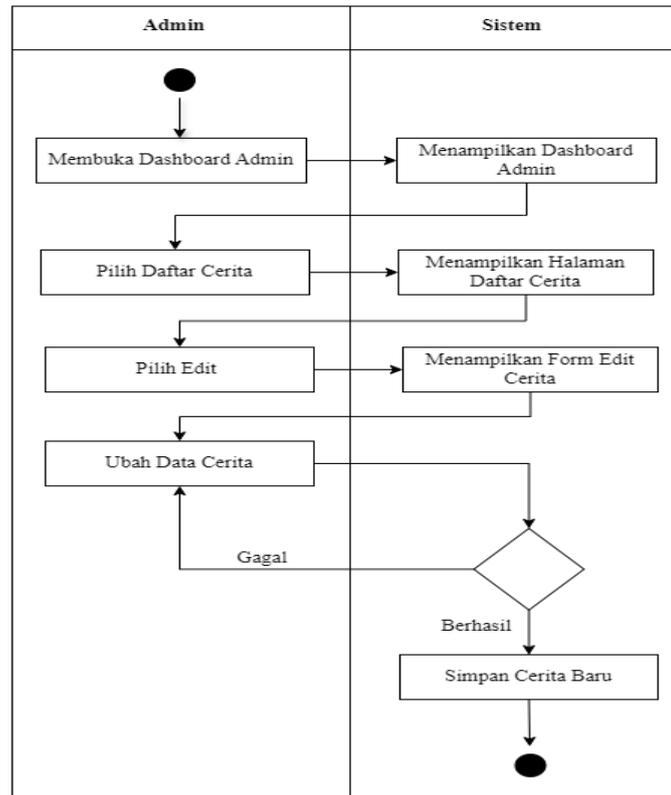
3) Activity Diagram Daftar Cerita

a) Activity Diagram Tambah Cerita



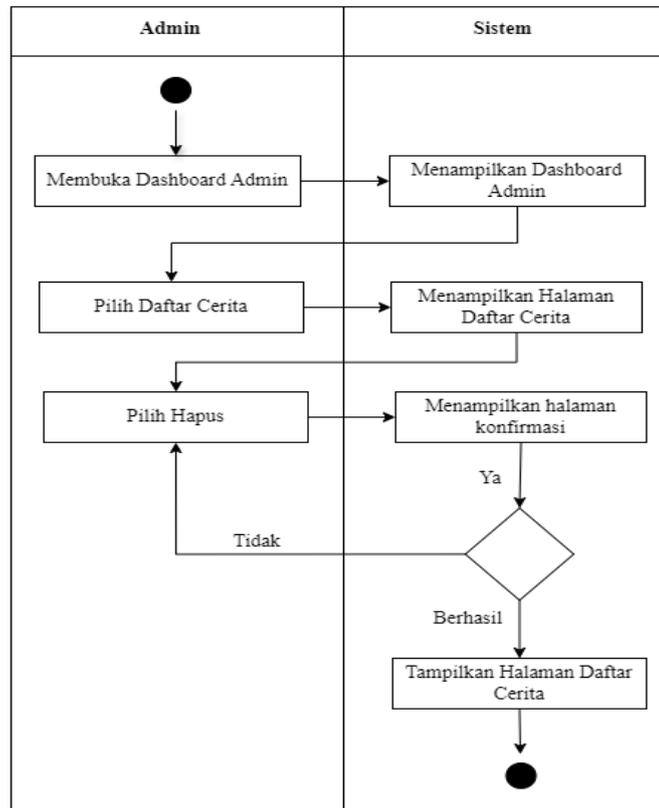
Gambar 4.3 Activity Diagram Tambah Cerita

Pada gambar diatas menjelaskan proses tambah cerita oleh admin. Pertama admin membuka halaman *dashboard* admin, kemudian sistem menampilkan halaman *dashboard* admin. Selanjutnya, admin pilih menu daftar cerita dan sistem menampilkan halaman daftar cerita. Untuk menambah data admin memilih menu tambah cerita, Maka sistem akan menampilkan form tambah cerita. Setelah itu admin mengisi form tambah cerita, setelah itu sistem akan melakukan validasi data, apabila berhasil maka sistem akan menampilkan halaman daftar cerita dan sebaliknya jika gagal maka sistem akan menampilkan pesan kesalahan dan admin diharapkan mengisi kembali form tambah cerita tersebut dengan benar.

b) *Activity Diagram* Edit Cerita

Gambar 4. 4 *Activity Diagram* Edit Cerita

Pada gambar diatas menjelaskan proses edit cerita oleh admin. Pertama admin membuka halaman *dashboard* admin, kemudian sistem menampilkan halaman *dashboard* admin. Selanjutnya, admin pilih menu daftar cerita dan sistem menampilkan halaman daftar cerita. Untuk mengedit data admin memilih menu edit, Maka sistem akan menampilkan form edit cerita. Setelah itu admin mengisi form edit cerita, setelah itu sistem akan melakukan validasi data, apabila berhasil maka sistem akan menampilkan halaman daftar cerita dan sebaliknya jika gagal maka sistem akan menampilkan pesan kesalahan dan admin diharapkan mengisi kembali form edit cerita tersebut dengan benar.

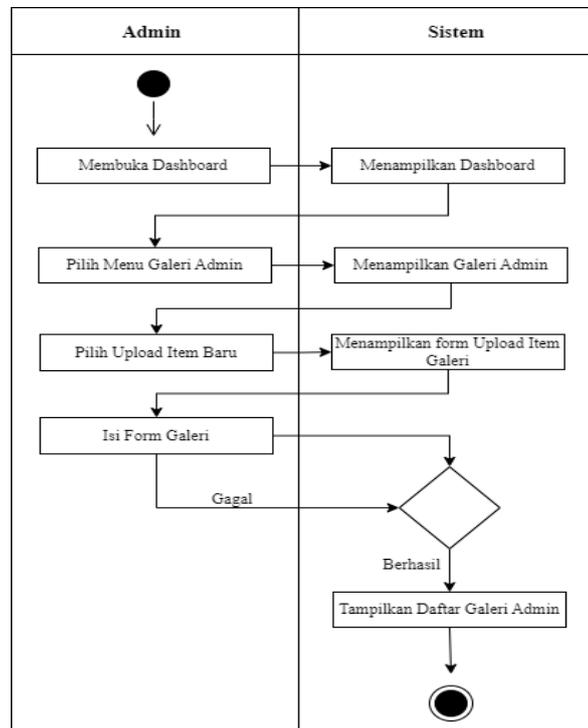
c) *Activity Diagram Hapus Cerita*

Gambar 4.5 *Activity Diagram Hapus Cerita*

Pada gambar diatas menjelaskan proses hapus cerita oleh admin. Pertama admin membuka halaman *dashboard* admin, kemudian sistem menampilkan halaman *dashboard* admin. Selanjutnya, admin pilih menu daftar cerita dan sistem menampilkan halaman daftar cerita. Untuk menghapus data admin memilih menu hapus, Maka sistem akan menampilkan halaman konfirmasi, setelah itu sistem akan melakukan validasi data, apabila berhasil maka sistem akan menampilkan halaman daftar cerita sebaliknya jika gagal maka sistem akan menampilkan pesan kesalahan dan admin diharapkan untuk menghapus cerita yang diinginkan tersebut dengan benar.

4) Activity Diagram Galeri Admin

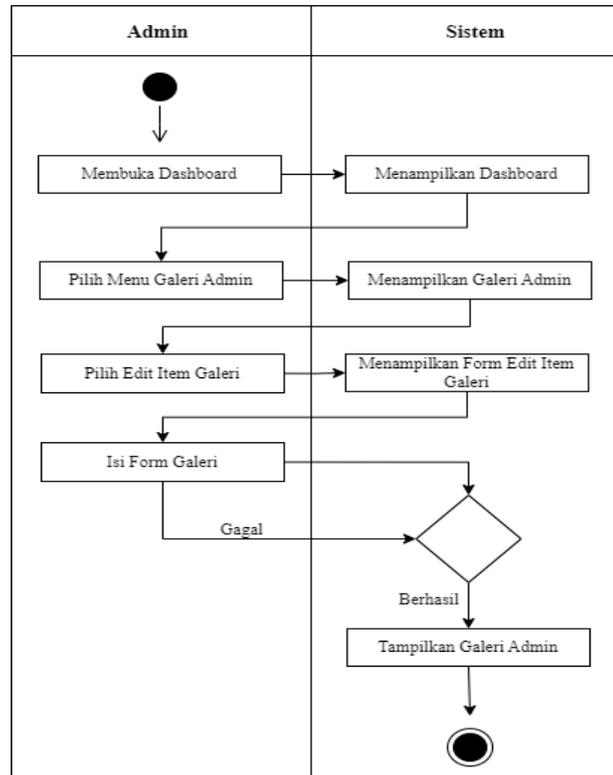
a. Activity Diagram Upload Item



Gambar 4. 6 Activity Diagram Upload Item

Pada gambar diatas menjelaskan proses *upload* item oleh admin. Pertama admin membuka halaman *dashboard* admin, kemudian sistem menampilkan halaman *dashboard* admin. Selanjutnya, admin pilih menu galeri admin dan sistem menampilkan halaman daftar galeri item. Untuk mengupload item baru admin memilih menu *upload* item baru, Maka sistem akan menampilkan *form upload* item baru. Setelah itu admin mengisi form galeri, setelah itu sistem akan melakukan validasi data, apabila berhasil maka sistem akan menampilkan halaman galeri admin dan sebaliknya jika gagal maka sistem akan menampilkan pesan kesalahan dan admin diharapkan mengisi kembali form *upload* item baru tersebut dengan benar.

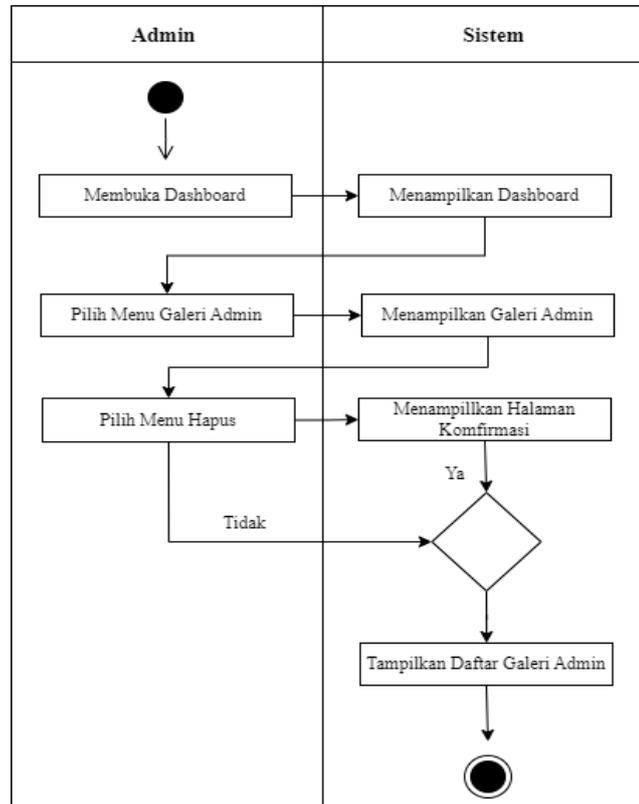
b. *Activity Diagram* Edit Item



Gambar 4. 7 *Activity Diagram* Edit Item

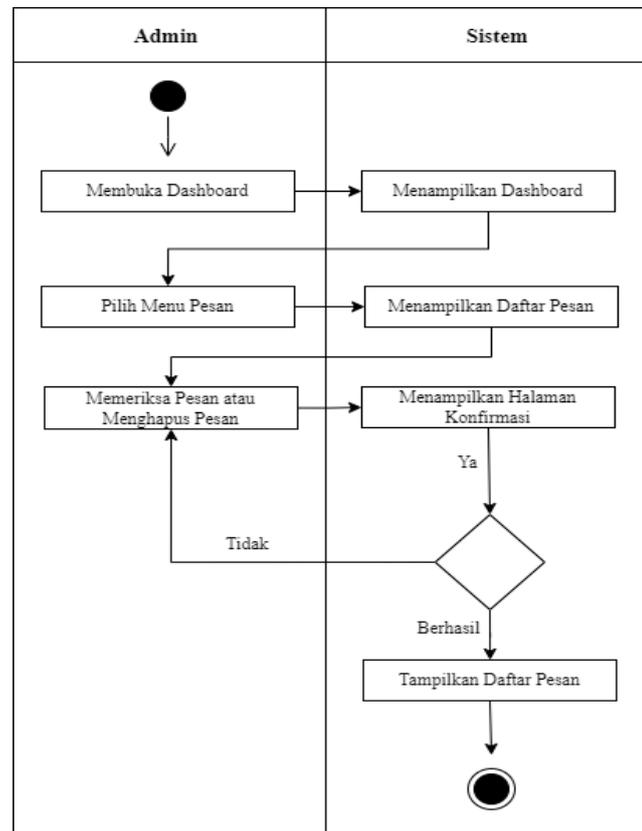
Pada gambar diatas menjelaskan proses edit item oleh admin. Pertama admin membuka halaman *dashboard* admin, kemudian sistem menampilkan halaman *dashboard* admin. Selanjutnya, admin pilih menu galeri admin dan sistem menampilkan halaman daftar galeri admin. Untuk mengedit data admin memilih menu edit, Maka sistem akan menampilkan form edit item. Setelah itu admin mengisi form edit item, setelah itu sistem akan melakukan validasi data, apabila berhasil maka sistem akan menampilkan halaman galeri admin yang ditampilkan data dan sebaliknya jika gagal maka sistem akan menampilkan pesan kesalahan dan admin diharapkan mengisi kembali form edit item tersebut dengan benar.

c. *Activity Diagram Hapus*



Gambar 4. 8 *Activity Diagram Hapus Item*

Pada gambar diatas menjelaskan proses hapus item oleh admin. Pertama admin membuka halaman *dashboard* admin, kemudian sistem menampilkan halaman *dashboard* admin. Selanjutnya, admin pilih menu galeri admin dan sistem menampilkan halaman daftar galeri admin. Untuk menghapus data admin memilih menu hapus, Maka sistem akan menampilkan halaman konfirmasi, setelah itu sistem akan melakukan validasi data, apabila berhasil maka sistem akan menampilkan halaman galeri admin sebaliknya jika gagal maka sistem akan menampilkan pesan kesalahan dan admin diharapkan untuk menghapus item galeri yang diinginkan tersebut dengan benar.

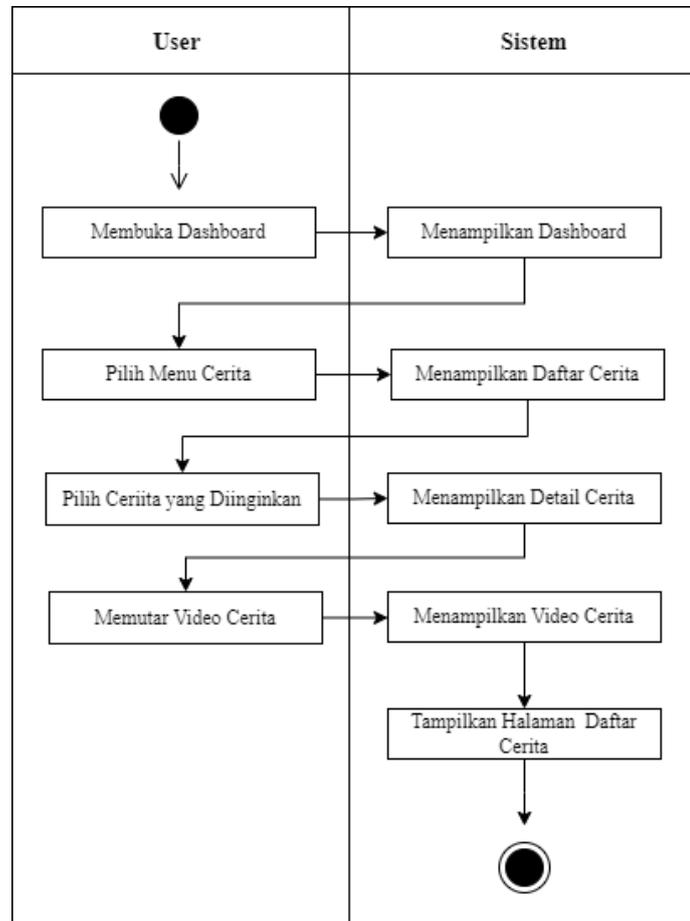
5) *Activity Diagram Pesan*

Gambar 4.9 *Activity Diagram Pesan*

Pada gambar diatas menjelaskan bagaimana proses pesan oleh admin. Pertama admin membuka halaman *dashboard* admin, kemudian sistem menampilkan halaman *dashboard* admin. Selanjutnya, admin pilih menu pesan dan sistem menampilkan halaman daftar pesan pengguna. Untuk menghapus admin memilih menu hapus, Maka sistem akan menampilkan halaman konfirmasi. setelah itu sistem akan melakukan validasi data, apabila berhasil maka sistem akan menampilkan halaman menu yang ditampilkan data dan sebaliknya jika gagal maka sistem akan menampilkan pesan kesalahan dan admin akan kembali ke halaman daftar pesan pengguna.

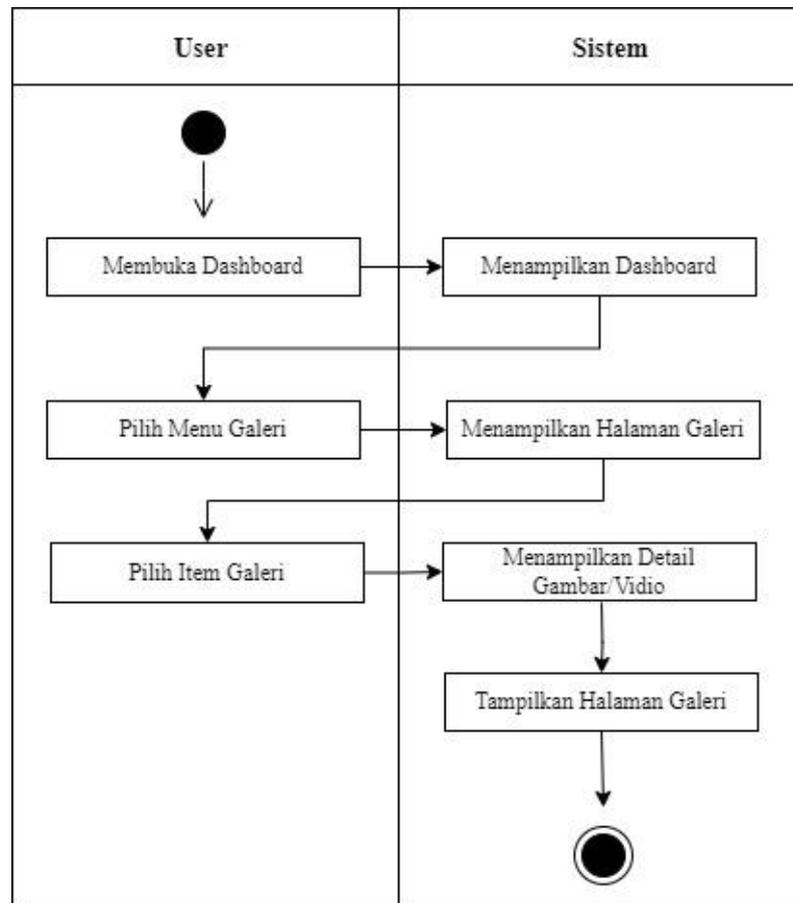
b. *Activity Diagram User*

1) *Activity Diagram Cerita*



Gambar 4. 10 *Activity Diagram Cerita*

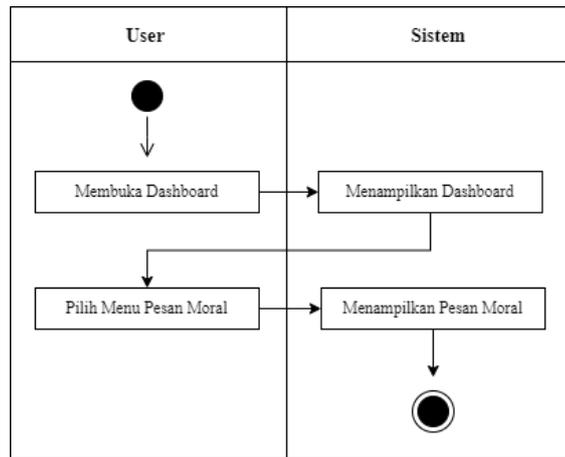
Pada gambar diatas menjelaskan bagaimana proses cerita oleh user. Pertama user membuka halaman *dashboard*, kemudian sistem menampilkan halaman *dashboard*. Selanjutnya, user pilih menu cerita dan sistem menampilkan halaman daftar cerita rakyat. User memilih cerita yang ingin dibaca atau ditonton. Sistem menampilkan detail cerita yang dipilih. User menonton animasi atau membaca cerita sesuai dengan pilihan yang dibuat. Setelah menonton atau membaca, user bisa kembali ke daftar cerita.

2) *Actiavity Diagram* Galeri

Gambar 4. 11 *Activity Diagram* Galeri

Pada gambar diatas menjelaskan bagaimana proses galeri oleh user. Pertama user membuka halaman *dashboard*, kemudian sistem menampilkan halaman *dashboard*. Selanjutnya, user pilih menu galeri dan Sistem menampilkan halaman galeri yang berisi gambar atau video. User memilih item galeri yang ingin dilihat. Sistem menampilkan detail gambar atau video yang dipilih. Setelah melihat atau menonton item galeri, user dapat kembali ke daftar galeri.

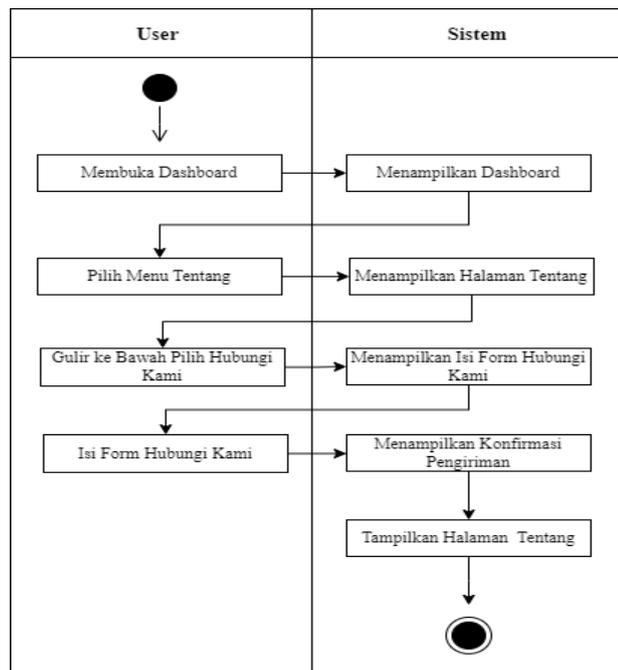
3) *Activity Diagram* Pesan Moral



Gambar 4. 12 *Activity Diagram* Pesan Moral

Pada gambar diatas menjelaskan bagaimana proses pesan moral oleh user. Pertama user membuka halaman *dashboard*, kemudian sistem menampilkan halaman *dashboard*. User pilih menu pesan moral dan Sistem menampilkan halaman pesan moral yang berisi pesan moral dari cerita rakyat.

4) *Activity Diagram* Tentang



Gambar 4. 13 *Activity Diagram* Tentang

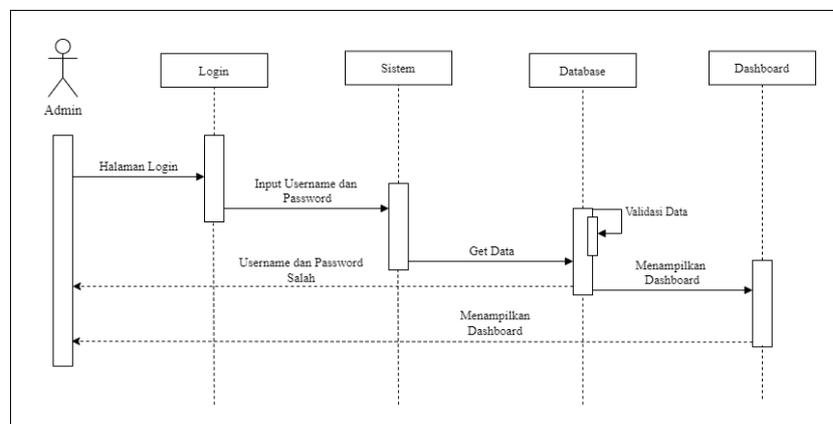
Pada gambar diatas menjelaskan bagaimana proses tentang oleh user. Pertama user membuka halaman *dashboard*, kemudian sistem menampilkan halaman *dashboard*. Selanjutnya, user pilih menu pesan tentang dan Sistem menampilkan halaman "Tentang Kami" yang berisi informasi terkait aplikasi dan tujuan. user halaman ke bawah dan menemukan bagian "Hubungi Kami" untuk menghubungi atau memberikan saran sistem kemudian menampilkan form. user mengisi form "Hubungi Kami" dengan data yang diperlukan seperti Nama, Email, Subject, dan Pesan. Sistem menampilkan pesan konfirmasi bahwa pesan telah berhasil dikirim.

2. Sequence Diagram

Sequence Diagram merupakan salah satu *Diagram Interaction* yang menjelaskan bagaimana suatu operasi itu dilakukan *message* (pesan) apa yang dikirim dan kapan pelaksanaannya.

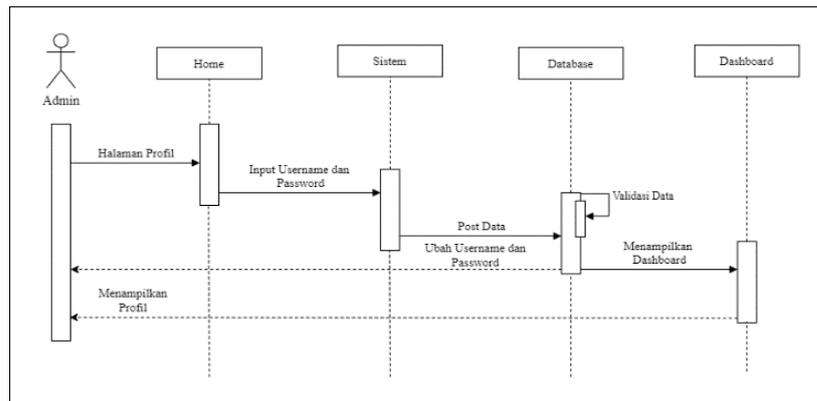
a. Sequence Diagram Admin

1. Sequence Diagram Login Admin



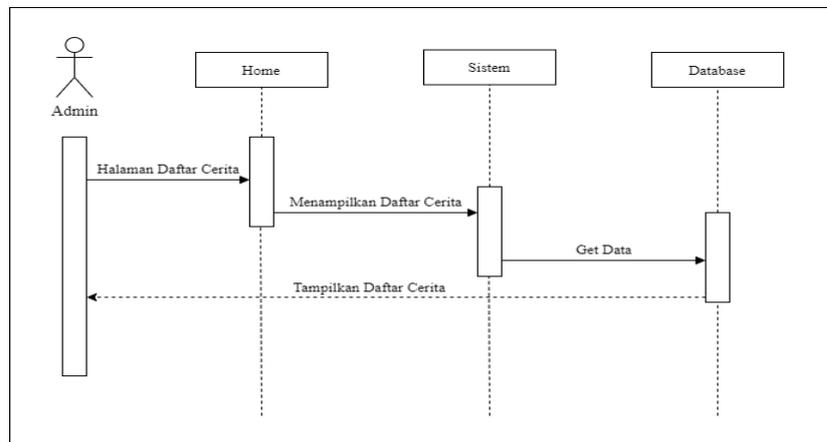
Gambar 4. 14 *Sequence Diagram Login Admin*

2. Sequence Diagram Profil Admin



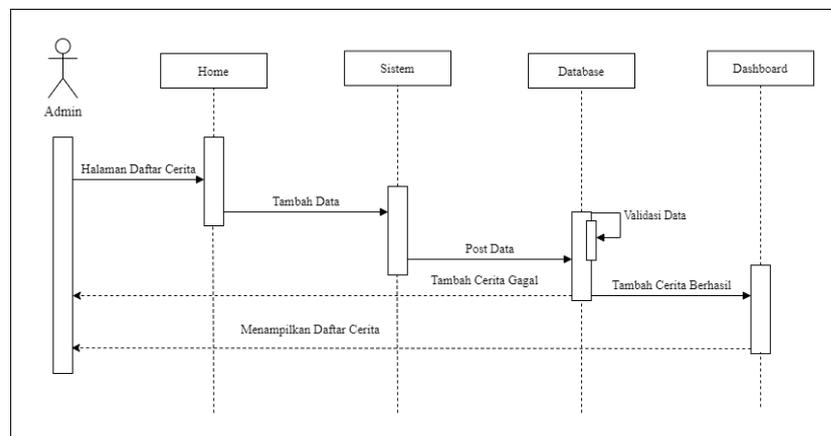
Gambar 4. 15 *Sequence Diagram* Profil Admin

3. *Sequence Diagram* Daftar Cerita

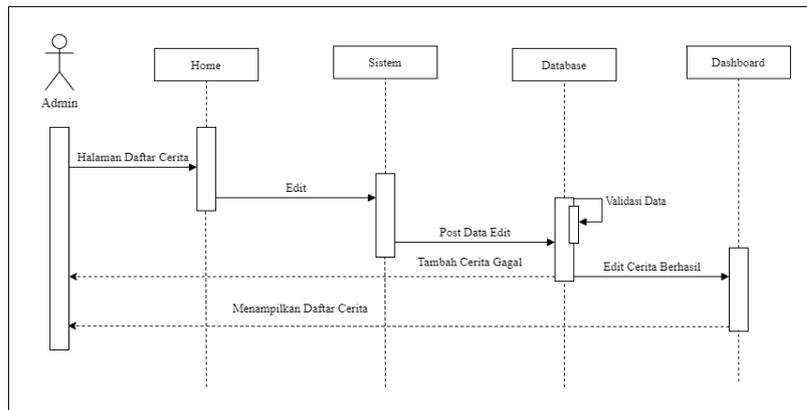
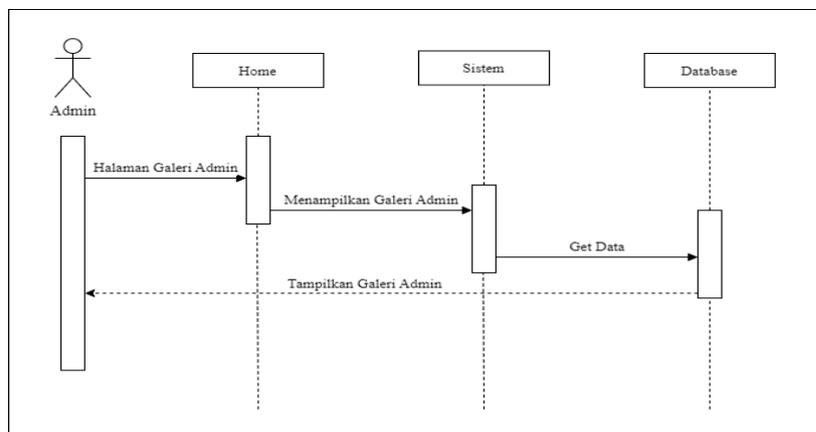
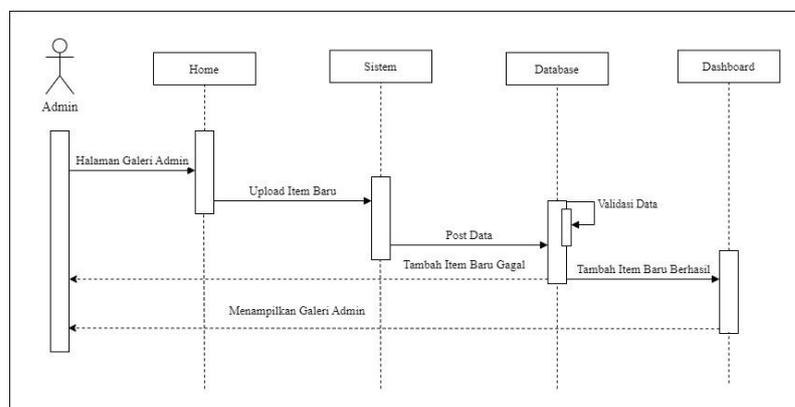


Gambar 4. 16 *Sequence Diagram* Daftar Cerita

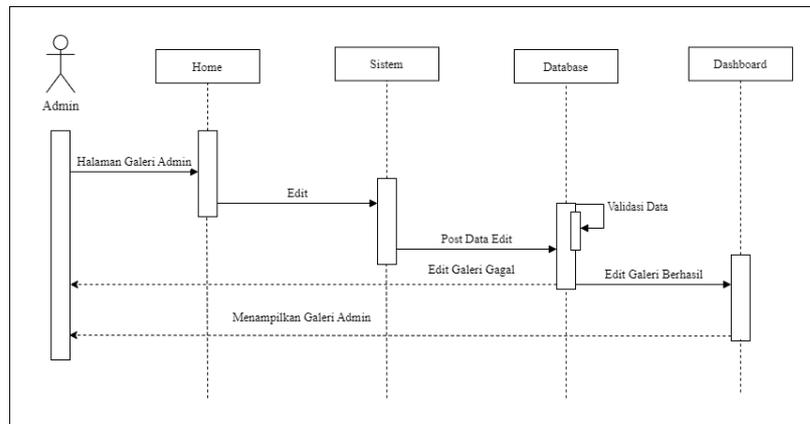
a) *Sequence Diagram* Tambah Cerita



Gambar 4. 17 *Sequence Diagram* Tambah Cerita

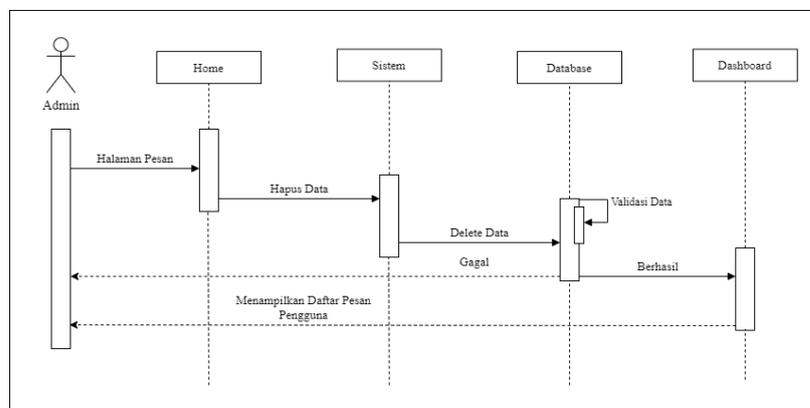
b) *Sequence Diagram Edit Cerita*Gambar 4. 18 *Sequence Diagram Edit Cerita*4. *Sequence Diagram Galeri Admin*Gambar 4. 19 *Sequence Diagram Galeri Admin*a) *Sequence Diagram Upload Item Baru*Gambar 4. 20 *Sequence Diagram Upload Item*

b) *Sequence Diagram* Edit Item Galeri



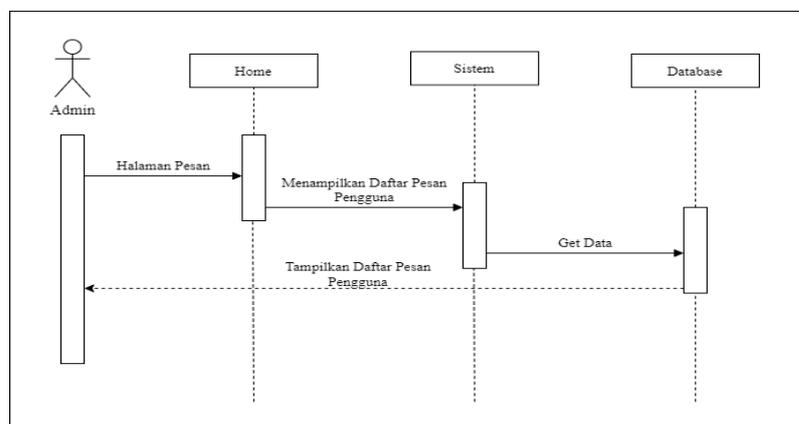
Gambar 4. 21 *Sequence Diagram* Edit Item

c) *Sequence Diagram* Hapus Item Galeri



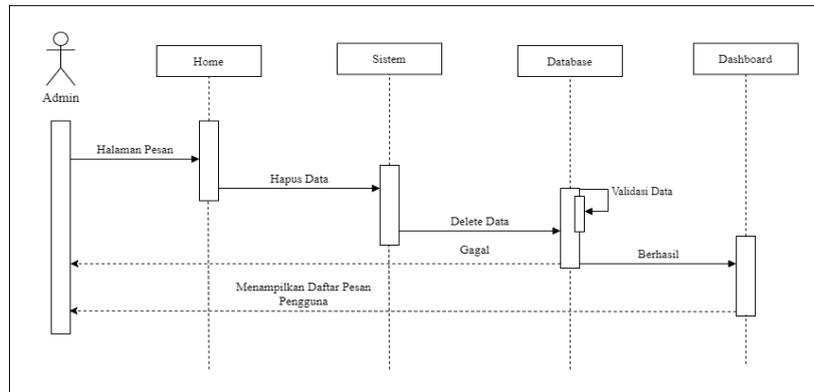
Gambar 4. 22 *Sequence Diagram* Hapus Item

5. *Sequence Diagram* Pesan



Gambar 4. 23 *Sequence Diagram* Pesan

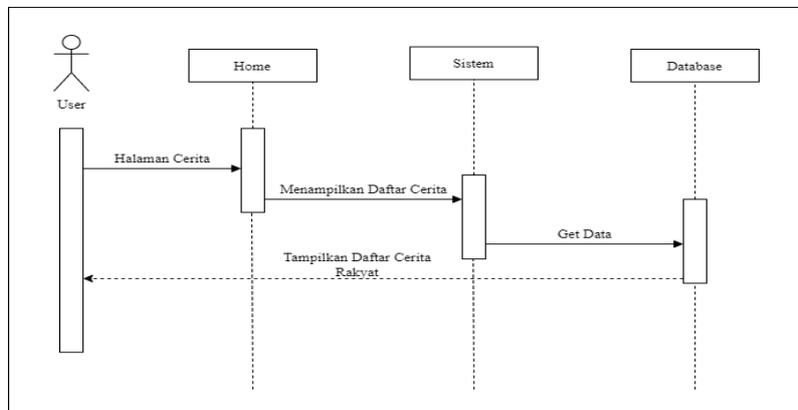
6. Sequence Diagram Hapus Pesan



Gambar 4. 24 Sequence Diagram Hapus Pesan

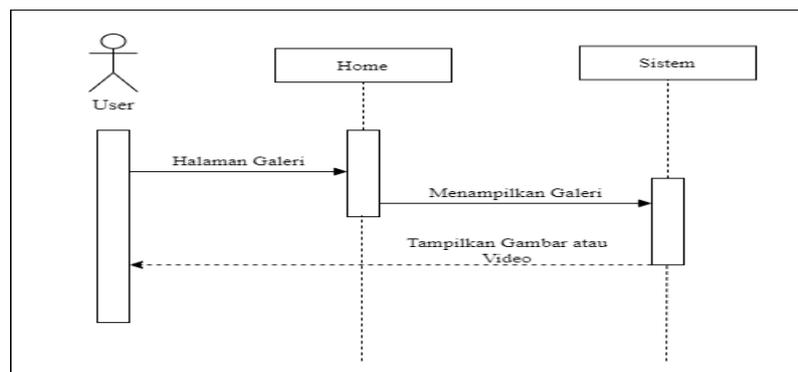
b. Sequence Diagram User

1. Sequence Diagram Cerita



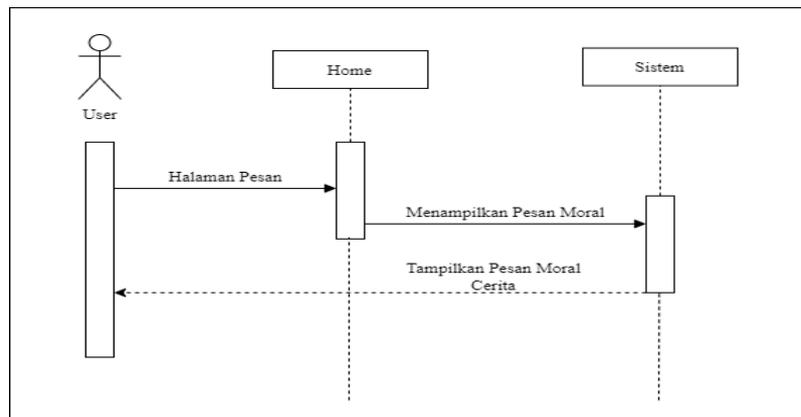
Gambar 4. 25 Sequence Diagram Cerita

2. Sequence Diagram Galeri



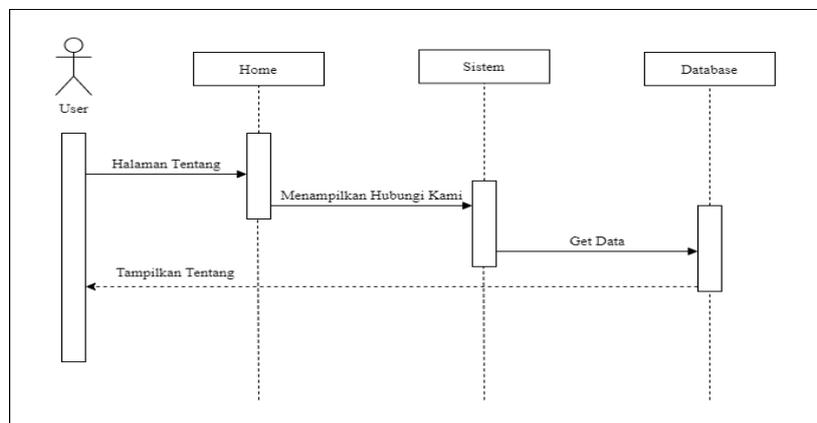
Gambar 4. 26 Sequence Diagram Galeri

3. Sequence Diagram Pesan Moral



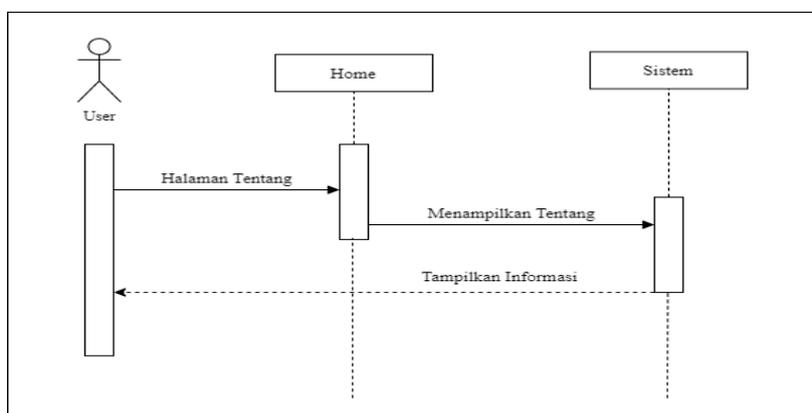
Gambar 4. 27 Sequence Diagram Pesan Moral

4. Sequence Diagram Tentang



Gambar 4. 28 Sequence Diagram Tentang

5. Sequence Diagram Hubungi



Gambar 4. 29 Sequence Diagram Hubungi

3. Kamus Data Database

Kamus data merupakan salah satu komponen penting dari DBMS yang mencakup informasi tentang struktur database. Setiap elemen data yang disimpan dalam database memiliki catatan kamus data yang menjelaskan elemen tersebut. Kamus data sangat membantu analisis sistem dalam mendefinisikan data yang mengalir dalam sistem, sehingga definisi data dapat dilakukan secara lengkap dan terstruktur. Berikut ini adalah kamus data dari aplikasi cerita rakyat.

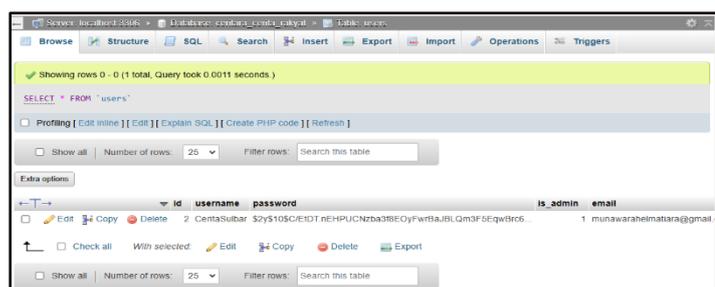
a. Kamus data *users*

Tabel 4. 1 Kamus Data *Users*

Nama Atribut	Tipe Data	Ukuran	Key
id	int	-	<i>Primary Key</i>
username	varchar	50	-
password	varchar	255	-
is_admin	tinyint	1	-
email	varchar	150	-

1) Tabel *User*

Tabel ini adalah tabel pengguna *users*, yang biasanya digunakan untuk menyimpan informasi terkait dengan pengguna aplikasi.



Gambar 4. 30 Tabel *User*

2) Membuat tabel *user* menggunakan perintah SQL untuk membuat tabel *users*:

```
CREATE TABLE users (
  id INT AUTO_INCREMENT PRIMARY KEY,
  username VARCHAR(50) NOT NULL,
  password VARCHAR(255) NOT NULL,
  is_admin TINYINT(1) NOT NULL DEFAULT 0,
  email VARCHAR(150) NOT NULL
);
```

Gambar 4. 31 *Codingan Tabel Users*

- a) *CREATE TABLE*: Perintah *SQL* untuk membuat tabel baru dalam database.
 - b) *users*: Nama tabel yang akan dibuat. Tabel ini dirancang untuk menyimpan data pengguna, sesuai konvensi penamaan umum.
 - c) *Id, username, password, is_admin, email*: Nama-nama kolom yang dibuat di dalam tabel *users*.
 - d) *NOT NULL*: Menandakan bahwa kolom tersebut wajib diisi. Jika tidak ada nilai yang diberikan untuk kolom yang memiliki atribut *NOT NULL*, maka database akan menolak memasukkan data tersebut.
- 3) Penjelasan kamus data *users* pada tabel. sebagai berikut:
- a) *Id* merupakan kolom id pengguna dari tabel *users* sekaligus sebagai *primary key*.
 - b) *Username* merupakan kolom berisi nama pengguna yang digunakan untuk login. Tipe datanya *varchar* yang memiliki panjang maksimum 50 karakter.
 - c) *Password* merupakan kolom berisi kata sandi untuk admin login. *Password* disimpan dalam bentuk hash (seperti *bcrypt*), jadi tipe datanya *varchar* dengan panjang teks 255 karakter.
 - d) *Is_admin* menyimpan informasi apakah pengguna adalah admin atau bukan. Tipe datanya *tinyint* memiliki nilai 1 untuk pengguna admin, 0 untuk pengguna biasa.

e) Email menyimpan alamat email pengguna. Tipe data varchar yang memiliki panjang teks 150 karakter.

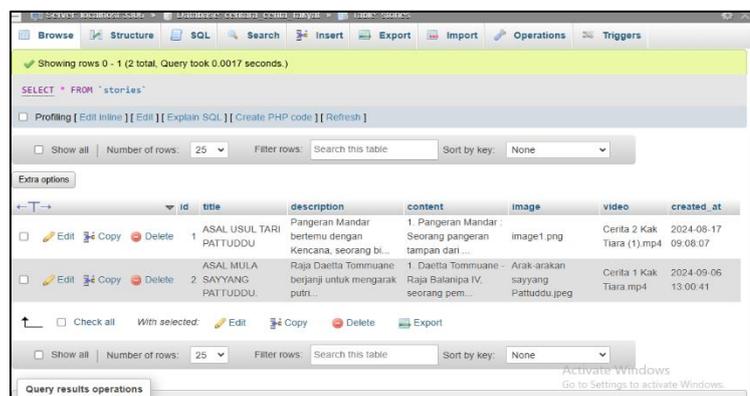
b. Kamus data *stories*

Tabel 4. 2 Kamus Data *Stories*

Nama Atribut	Tipe Data	Ukuran	Key
id	int	-	Primary Key
title	varchar	255	-
description	text	-	-
content	text	-	-
image	varchar	255	-
video	varchar	255	-
create_at	timestamp	-	-

1) tabel *stories*

Tabel ini digunakan untuk mengelola data cerita dalam sebuah aplikasi seperti menyimpan data tentang cerita, termasuk judul, deskripsi, konten, dan multimedia (gambar/video).



The screenshot shows a database management interface with the following data in the 'stories' table:

id	title	description	content	image	video	created_at
1	ASAL USUL TARI PATUDDU	Pangeran Mandar bertemu dengan Kencana, seorang bi...	1. Pangeran Mandar: Seorang pangeran tampan dari ...	image1.png	Cerita 2 Kak Tiara (1).mp4	2024-08-17 09:08:07
2	SAYYANG PATUDDU.	Raja Daetta Tommuane bejanji untuk mengarak putri...	1. Daetta Tommuane - Raja Balanipa IV, seorang pem...	Arak-arakan sayyang Pattuddu.jpeg	Cerita 1 Kak Tiara.mp4	2024-09-06 13:00:41

Gambar 4. 32 Tabel *Stories*

2) Membuat tabel *stories* menggunakan perintah SQL untuk membuat tabel

stories:

```
CREATE TABLE gallery_items (
  id INT AUTO_INCREMENT PRIMARY KEY,
  title VARCHAR(255) NOT NULL,
  description TEXT,
  content TEXT,
  image VARCHAR(255),
  video VARCHAR(255),
  create_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

Gambar 4. 33 *Codingan* Tabel *Stories*

- a) *CREATE TABLE*: Perintah SQL untuk membuat tabel baru dalam database.
 - b) *stories*: Nama tabel yang akan dibuat. Tabel ini dirancang untuk menyimpan data pengguna, sesuai konvensi penamaan umum.
 - c) *Id*, *title*, *description*, *content*, *image*, *video*, dan *create_at* : Nama-nama kolom yang dibuat di dalam tabel *stories*.
 - d) NOT NULL: Menandakan bahwa kolom tersebut wajib diisi. Jika tidak ada nilai yang diberikan untuk kolom yang memiliki atribut NOT NULL, maka database akan menolak memasukkan data tersebut.
- 3) Penjelasan kamus data *stories* pada tabel. sebagai berikut:
- a) *Id* merupakan kolom id pengguna dari tabel *stories* sekaligus sebagai *primary key*.
 - b) *Title* merupakan kolom berisi judul cerita, menggunakan tipe data *varchar* yang memiliki panjang teks 255 karakter.
 - c) *Description* merupakan kolom berisi deskripsi singkat cerita, menggunakan tipe data *text* yang memungkinkan penyimpanan teks panjang cukup besar.

- d) *Content* merupakan kolom berisi konten lengkap dari cerita. menggunakan tipe data text yang memungkinkan penyimpanan text dengan panjang yang cukup besar.
- e) *Image* merupakan kolom berisi file gambar *thumbnail* terkait cerita, menggunakan tipe data varchar yang membatasi panjang teks hingga 255 karakter.
- f) *Video* merupakan kolom berisi file video terkait cerita, menggunakan tipe data varchar yang membatasi panjang teks hingga 255 karakter.
- g) *Create_at* merupakan kolom berisi waktu dan tanggal ketika cerita dibuat, dengan tipe data timestamp karena diisi otomatis saat pembuatan.
- c. Kamus data *gallery_items*

Tabel 4. 3 Kamus Data *Gallery_items*

Nama Atribut	Tipe Data	Ukuran	Key
id	int	-	<i>Primary Key</i>
title	varchar	255	-
media_type	enum	-	-
file_path	varchar	255	-
category	enum	-	-
create_at	timestamp	-	-

1) Tabel *gallery_items*

Tabel ini digunakan untuk mengelola data galeri dalam sebuah aplikasi seperti menyimpan informasi detail tentang item media, termasuk jenis media, judul, kategori, dan file path.

- 2) Membuat tabel *gallery_items* menggunakan perintah SQL untuk membuat tabel *gallery_items*.

```
CREATE TABLE gallery_items (
  id INT AUTO_INCREMENT PRIMARY KEY,
  title VARCHAR(255) NOT NULL,
  media_type ENUM('image', 'video', 'audio') NOT NULL,
  file_path VARCHAR(255) NOT NULL,
  category ENUM('news', 'entertainment', 'education') NOT NULL,
  create_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

Gambar 4. 34 Codingan Tabel *Galley_items*

- a) *CREATE TABLE*: Perintah SQL untuk membuat tabel baru dalam database.
 - b) *gallery_items*: Nama tabel yang akan dibuat. Tabel ini dirancang untuk menyimpan data pengguna, sesuai konvensi penamaan umum.
 - c) *Id*, *title*, *media_type*, *file_path*, *category*, dan *create_at* : Nama-nama kolom yang dibuat di dalam tabel *stories*.
 - d) NOT NULL: Menandakan bahwa kolom tersebut wajib diisi. Jika tidak ada nilai yang diberikan untuk kolom yang memiliki atribut NOT NULL, maka database akan menolak memasukkan data tersebut.
- 3) Penjelasan kamus data *gallery_items* pada tabel. sebagai berikut:
- 1) *Id* merupakan kolom id pengguna dari tabel *gallery_items* sekaligus sebagai *primary key*.
 - 2) *Title* merupakan kolom yang menyimpan judul dari item galeri. Menggunakan tipe data varchar untuk menyimpan teks dengan panjang yang dapat bervariasi.
 - 3) *Media_type* digunakan untuk menentukan media dari item yang sudah diupload. Menggunakan tipe data enum untuk membatasi nilai yang dapat

disimpan di dalam kolom hanya bisa menyimpan salah satu dari dua nilai: *'image'* atau *'video'*.

- 4) *File_path* merupakan kolom yang digunakan untuk menyimpan lokasi dari file media yang diupload. Tipe data varchar menyimpan teks hingga 255 karakter.
- 5) *Category* merupakan kolom yang digunakan untuk membantu mengelompokkan dari item galeri. Tipe data yang digunakan enum untuk mendefinisikan beberapa kategori misalnya, tari pattuddu atau kategori lain yang sesuai dengan kebutuhan.
- 6) *Create_at* merupakan kolom berisi waktu dan tanggal ketika cerita dibuat, dengan tipe data timestamp karena diisi otomatis saat pembuatan.

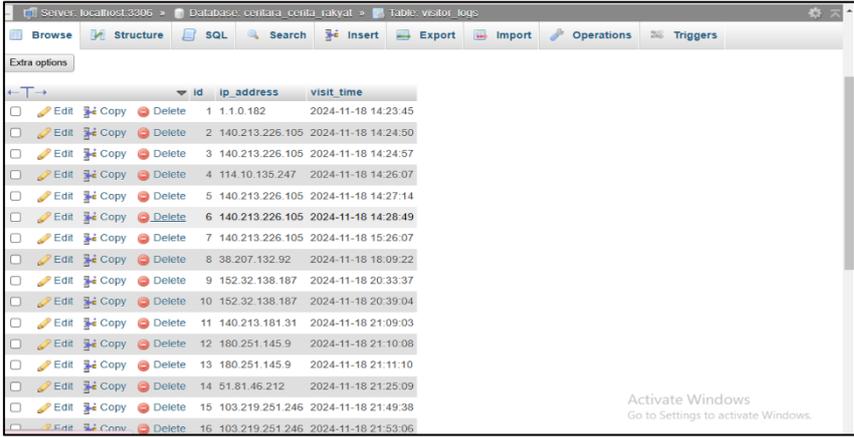
d. Kamus data *visitor_logs*

Tabel 4. 4 Kamus data *visitor_logs*

Nama Atribut	Tipe Data	Ukuran	Key
id	int	-	<i>Primary Key</i>
Ip_address	varchar	45	-
Visit time	timestamp	-	-

1) Database *visitor_logs*

Tabel ini digunakan untuk mencatat dan melacak aktivitas kunjungan (visitor log) ke suatu situs web. Data ini biasanya mencakup alamat IP pengunjung dan waktu kunjungan.



	id	ip_address	visit_time
<input type="checkbox"/>	1	1.1.0.182	2024-11-18 14:23:45
<input type="checkbox"/>	2	140.213.226.105	2024-11-18 14:24:50
<input type="checkbox"/>	3	140.213.226.105	2024-11-18 14:24:57
<input type="checkbox"/>	4	114.10.135.247	2024-11-18 14:26:07
<input type="checkbox"/>	5	140.213.226.105	2024-11-18 14:27:14
<input type="checkbox"/>	6	140.213.226.105	2024-11-18 14:28:49
<input type="checkbox"/>	7	140.213.226.105	2024-11-18 15:26:07
<input type="checkbox"/>	8	38.207.132.92	2024-11-18 18:09:22
<input type="checkbox"/>	9	152.32.138.187	2024-11-18 20:33:37
<input type="checkbox"/>	10	152.32.138.187	2024-11-18 20:39:04
<input type="checkbox"/>	11	140.213.181.31	2024-11-18 21:09:03
<input type="checkbox"/>	12	180.251.145.9	2024-11-18 21:10:08
<input type="checkbox"/>	13	180.251.145.9	2024-11-18 21:11:10
<input type="checkbox"/>	14	51.81.46.212	2024-11-18 21:25:09
<input type="checkbox"/>	15	103.219.251.246	2024-11-18 21:49:38
<input type="checkbox"/>	16	103.219.251.246	2024-11-18 21:53:06

Gambar 4. 35 Tabel *Visitor_logs*

- 2) Membuat tabel *visitor_logs* menggunakan perintah SQL untuk membuat tabel *visitor_logs*.

```
CREATE TABLE visitor_logs (
  id INT AUTO_INCREMENT PRIMARY KEY,
  ip_address VARCHAR(45) NOT NULL,
  visit_time DATETIME NOT NULL DEFAULT NOW()
);
```

Gambar 4. 36 Codingan Tabel *Visitor_logs*

- a) *CREATE TABLE*: Perintah SQL untuk membuat tabel baru dalam database.
 - b) *visitor_logs*: Nama tabel yang akan dibuat. Tabel ini dirancang untuk menyimpan data pengguna, sesuai konvensi penamaan umum.
 - c) *Id*, *Ip_address*, dan *Visit_time*: Nama-nama kolom yang dibuat di dalam tabel *stories*.
 - d) *NOT NULL*: Menandakan bahwa kolom tersebut wajib diisi. Jika tidak ada nilai yang diberikan untuk kolom yang memiliki atribut *NOT NULL*, maka database akan menolak memasukkan data tersebut.
- 3) Penjelasan kamus data *visitor_logs* pada tabel. sebagai berikut:
- a) *Id* merupakan kolom id pengguna dari tabel *visitor_logs* sekaligus sebagai *primary key*.

- b) Ip Address merupakan kolom untuk menyimpan alamat IP dari pengunjung yang mengakses aplikasi. tipe data varchar dipilih karena alamat IP biasanya memiliki panjang maksimal sekitar 15 karakter.
- c) *Visit time* merupakan kolom mencatat waktu kunjungan setiap kali pengunjung mengakses aplikasi. tipe data timestamp karena diisi otomatis saat pembuatan.
- e. Kamus data *contact_messages*

Tabel 4. 5 Kamus Data *contact_messages*

Nama Atribut	Tipe Data	Ukuran	Key
id	int	-	<i>Primary Key</i>
name	varchar	100	-
email	varchar	100	-
subject	varchar	250	-
message	text	-	-
create_at	timestamp	-	-

1) Tabel *contact_messages*

Tabel *contact_messages* untuk menyimpan informasi lengkap terkait pesan yang dikirimkan oleh pengguna tabel ini memungkinkan sistem untuk mengelola komunikasi antara pengguna dan admin secara efektif. Data yang disimpan juga dapat digunakan untuk tindak lanjut, pelaporan, atau analisis komunikasi pelanggan.

	id	name	email	subject	message	created_at
<input type="checkbox"/>	1	Nayra	Nayraalnayra05@gmail.com	sayang patuudu	Ceritanya sangat menarik sehingga kita tidak perlu...	2024-09-28 22:51:40
<input type="checkbox"/>	2	Rany	ranywindr19@gmail.com	langgapan	wii keren E!!!	2024-11-21 16:58:02

Gambar 4. 37 Tabel *contact_messages*

- 2) Membuat tabel *contact_messages* menggunakan perintah SQL untuk membuat tabel *contact_messages*.

```
CREATE TABLE contact_messages (
  id INT AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(100) NOT NULL,
  email VARCHAR(100) NOT NULL,
  subject VARCHAR(250) NOT NULL,
  message TEXT NOT NULL,
  create_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

Gambar 4. 38 Codingan Tabel *Contact_messages*

- a) *CREATE TABLE*: Perintah SQL untuk membuat tabel baru dalam database.
 - b) *contact_messages* : Nama tabel yang akan dibuat. Tabel ini dirancang untuk menyimpan data pengguna, sesuai konvensi penamaan umum.
 - c) *Id, name, email, subject, message, dan create_at*: Nama-nama kolom yang dibuat di dalam tabel *stories*.
 - d) *NOT NULL*: Menandakan bahwa kolom tersebut wajib diisi. Jika tidak ada nilai yang diberikan untuk kolom yang memiliki atribut *NOT NULL*, maka database akan menolak memasukkan data tersebut.
- 3) Penjelasan kamus data *contact_messages* pada tabel. sebagai berikut:
- a) *Id* merupakan kolom id pengguna dari tabel *contact_messages* sekaligus sebagai *primary key*.

- b) *Name* merupakan kolom berisi nama untuk menyimpan nama pengirim, menggunakan tipe data varchar untuk menyimpan teks dengan panjang variabel 100 karakter.
 - c) *Email* merupakan kolom berisi alamat email untuk menyimpan nama pengirim, menggunakan tipe data varchar untuk menyimpan teks dengan panjang variabel 100 karakter.
 - d) *Subject* merupakan untuk menyimpan subjek pesan, menggunakan tipe data varchar untuk menyimpan teks panjang dengan variabel 100 karakter.
 - e) *Message* merupakan kolom berisi pesan yang dikirimkan, menggunakan tipe data text digunakan untuk menyimpan teks panjang
 - f) *Create_at* merupakan kolom berisi Waktu dan tanggal ketika cerita dibuat, dengan tipe data timestamp karena diisi otomatis saat pembuatan.
- f. Koneksi Database

Koneksi database digunakan menghubungkan aplikasi dengan database MySQL menggunakan parameter konfigurasi seperti host, nama database, username, dan password.

```

<?php
$host = 'localhost';
$db = 'cerita_rakyat';
$user = 'root';
$pass = '';
try {
    $pdo = new PDO("mysql:host=$host;dbname=$db", $user, $pass);
    $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
} catch (PDOException $e) {
    echo "Error: " . $e->getMessage();
}
?>

```

Gambar 4. 39 Code Koneksi ke Database

1. Penjelasan Kode Koneksi Database

Konfigurasi Database: Variabel **\$host**: Menentukan nama host server tempat database berada. **\$db**: Nama database yang ingin diakses. **\$user**: Nama pengguna database. **\$pass**: Kata sandi pengguna database. **PDO**: menghubungkan aplikasi ke database. "**mysql:host=\$host; dbname=\$db**": Membuat string koneksi ke database MySQL. **\$user dan \$pass**: Digunakan untuk autentikasi ke database.

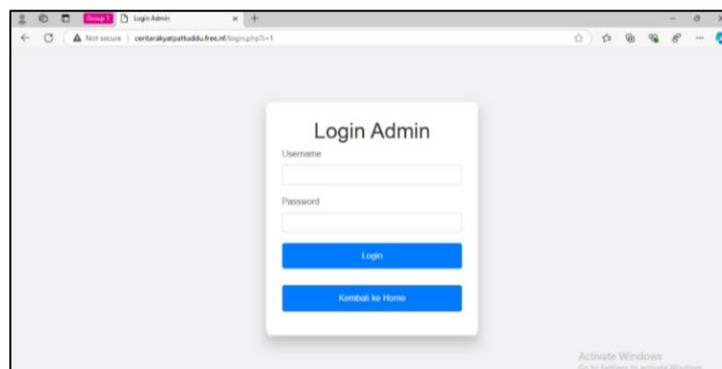
PDO::ATTR_ERRMODE: Mengatur mode kesalahan dari koneksi database. **PDO::ERRMODE_EXCEPTION**: Menyatakan bahwa jika terjadi kesalahan dalam koneksi, sebuah *exception* (error) akan dilemparkan. **try**: Blok ini mencoba menjalankan kode koneksi database menggunakan PDO.

B. Detail Sistem

1. Admin

a. Halaman *Login*

Merupakan halaman awal yang digunakan admin untuk mendapat akses masuk ke halaman menu admin.



Gambar 4. 40 Halaman *Login*

```

<div class="login-container">
  <h1>Login Admin</h1>
  <?php if (isset($error)): ?>
    <div class="alert alert-danger"><?= htmlspecialchars($error) ?></div>
  <?php endif; ?>
  <form method="POST" action="login.php">
    <div class="form-group">
      <label for="username">Username</label>
      <input type="text" class="form-control" id="username" name="username" required>
    </div>
    <div class="form-group">
      <label for="password">Password</label>
      <input type="password" class="form-control" id="password" name="password"
required>
    </div>
    <button type="submit" class="btn btn-primary">Login</button>
  </form>

  <div class="mt-3">
    <a href="index.php" class="btn btn-secondary">Kembali ke Home</a>
  </div>
</div>

```

Gambar 4. 42 *Codingan Halaman Login*

```

.login-container {
  background: #fff;
  padding: 30px;
  border-radius: 10px;
  box-shadow: 0 15px 25px rgba(0, 0, 0, 0.2);
  width: 100%;
  max-width: 400px;
  text-align: center;
  position: absolute;
  top: 50%;
  left: 50%;
  transform: translate(-50%, -50%);
}
.login-container h2 {
  font-size: 2rem;
  margin-bottom: 20px;
  color: #333;
}
.login-container .form-group {
  margin-bottom: 20px;
  text-align: left;
}
.login-container .form-group label {
  font-size: 1rem;
  color: #666;
}
.login-container .form-control {
  border-radius: 5px;
  border: 1px solid #ddd;
  padding: 10px;
  font-size: 1rem;
  width: 100%;
}
.login-container .btn {
  background-color: #007bff;
  color: #fff;
  font-size: 1rem;
  padding: 12px 0;
  border-radius: 5px;
  width: 100%;
  margin-bottom: 15px;
}

```

Gambar 4. 42 *CSS Halaman Login*

1. Penjelasan Kode Halaman *Login*

login-container: Div utama yang menjadi seluruh elemen di halaman login. Gaya untuk elemen ini didefinisikan di CSS pada bagian *.login-container*. Judul halaman login dengan teks "Login Admin". Ini memberikan informasi kepada pengguna bahwa ini adalah halaman untuk masuk sebagai admin.

PHP digunakan untuk menampilkan pesan *error* jika terdapat kesalahan, seperti login gagal. *\$error* adalah variabel yang berisi pesan error (misalnya, username atau password salah). Fungsi *htmlspecialchars()* memastikan keamanan dengan mencegah serangan XSS (*Cross-Site Scripting*) dengan menghindari interpretasi HTML pada isi pesan.

Formulir dengan metode *POST*, yang akan mengirimkan data ke file *login.php* untuk diproses. Tampilan Input *username* terdiri dari `<label>`: Label untuk input *username*. `<input type="text">`: Bidang input untuk *username*, wajib diisi. `class="form-control"`: Menggunakan kelas *Bootstrap* untuk gaya input agar terlihat rapi dan seragam. Tampilan input *password* terdiri dari `<label>`: Label untuk input *password*. `<input type="password">`: Bidang input untuk *password*, dengan tipe *password* untuk menyembunyikan karakter yang diketik.

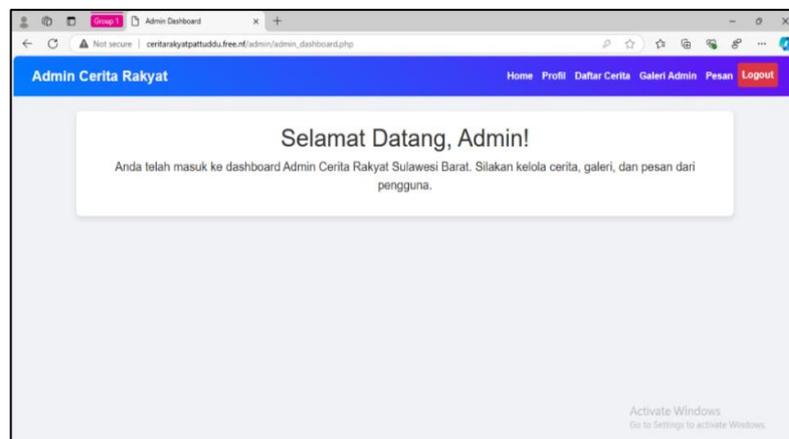
Tombol *submit* dengan teks "*Login*". `class="btn btn-primary"`: Menggunakan gaya *Bootstrap* untuk menampilkan tombol berwarna biru dengan efek hover. Tombol kembali ke halaman utama (*Home*) menggunakan tautan ke *index.php*. `class="btn btn-secondary"`: Menggunakan gaya *Bootstrap* untuk membuat tombol abu-abu.

Login-container pada CSS Ini memusatkan dan merapikan container utama. Sesuai dengan kebutuhan pada kode HTML `<div class="login-container">`. Judul (h1 atau h2) teks diberi warna #333 (abu-abu tua) dan jarak bawah 20px. Perubahan ini tetap berlaku jika diganti dengan `<h1>`. *Form Group* Setiap kelompok form (label dan *input*) memiliki jarak bawah 20px dan disejajarkan ke kiri (*text-align: left*). Ini sesuai untuk elemen `<div class="form-group">`.

Label ukuran teks label 1rem dengan warna #666 (abu-abu sedang). Ini cocok dengan elemen label `<label>` dalam kode HTML. *Input field* terlihat rapi dan seragam dengan gaya *Bootstrap*: Sudut melengkung (*border-radius: 5px*), lebar penuh container (*width: 100%*) dan sesuai dengan elemen `<input>` untuk *username* dan *password*. Tombol login memiliki: warna biru terang (*background-color: #007bff*) dan teks putih (*color: #fff*), mengisi seluruh lebar container (*width: 100%*) dan sudut melengkung (*border-radius: 5px*).

b. Halaman *Dashboard* Admin

Merupakan halaman awal admin yang berisi selamat datang ketika admin berhasil login.



Gambar 4. 43 Halaman *Dashboard* Admin

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Admin Dashboard</title>
  <link rel="stylesheet" href="../css/admin_style.css">
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<body>
  <?php include 'navbar_admin.php'; ?>
  <div class="container mt-2">
    <div class="row">
      <div class="col-12 text-center">
        <h1 class="welcome-message">Selamat Datang, Admin!</h1>
        <p class="lead">Anda telah masuk ke dashboard admin Cerita Rakyat Sulawesi Barat. Silahkan kelola cerita rakyat, galeri, dan pesan dari pengguna </p>
      </div>
    </div>
  </div>
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>

```

Gambar 4. 44 *Codingan Halaman Dashboard Admin*

```

.welcome-message {
  font-size: 1.3rem;
  font-weight: bold;
  color: #007bff;
  text-align: center;
  margin-bottom: 10px;
}
.lead {
  font-size: 0.9rem;
  color: #6c757d;
  text-align: center;
}

```

Gambar 4. 45 *CSS Halaman Dashboard Admin*

1. Penjelasan Kode Halaman *Dashborad* Admin

DOCTYPE dan Tag HTML: Menentukan bahwa halaman menggunakan HTML5. Meta tag: `charset="UTF-8"` memastikan semua karakter dapat diinterpretasikan. *Viewport* membuat tampilan responsif di perangkat yang berbeda (desktop atau mobile). *Title*: Judul halaman di tab browser (Admin Dashboard). *File* `navbar_admin.php` dimuat untuk membuat menu navigasi di bagian atas halaman. Isi `navbar_admin.php`: Biasanya mencakup link seperti "*Home*," "*Profil*," "*Logout*." dll.

Pada halaman ini *Bootstrap Grid System* digunakan *container*: Membuat area konten terpusat dengan margin otomatis. *row*: Baris untuk menyusun elemen (horizontal). *col-12*: Kolom penuh (100%) untuk menyusun teks di tengah. *text-center*: Mengatur teks menjadi rata tengah. Elemen *Teks*: `<h1>`: Menampilkan pesan selamat datang dengan teks besar dan `<p>`: Menampilkan deskripsi singkat dengan ukuran teks lebih kecil. *Script JavaScript Bootstrap* Memuat pustaka *JavaScript* untuk mendukung fungsi interaktif *Bootstrap*, seperti *dropdown*, *modal*, atau *navbar* yang bisa digeser.

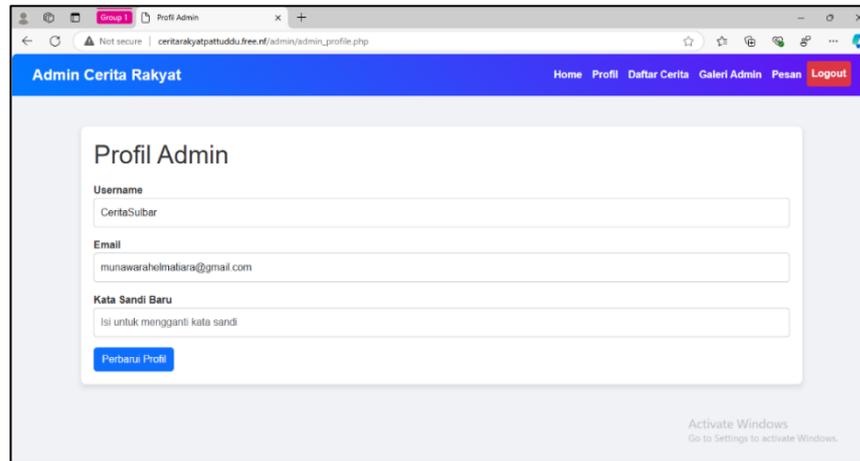
File CSS dapat digunakan untuk mengubah warna, *margin*, atau *font*. Untuk *welcome message* mengatur tampilan untuk elemen yang memiliki *class welcome-message* dalam HTML. Properti yang digunakan: *font-size*: 1.3rem; Mengatur ukuran *font* sebesar 1.3 rem (unit relatif terhadap ukuran font elemen root). *font-weight*: *bold*; Membuat teks menjadi tebal. *color*: #007bff; Memberikan warna biru dengan kode heksadesimal #007bff. *text-align*: *center*; Mengatur teks agar rata tengah. *margin-bottom*: 10px; Memberikan jarak vertikal sebesar 10 piksel di bawah elemen tersebut.

Mengatur elemen HTML dengan *class lead*. Properti yang digunakan: *font-size*: 0.9rem; Mengatur ukuran *font* lebih kecil, yaitu 0.9 rem. *color*: #6c757d; Memberikan warna abu-abu dengan kode heksadesimal #6c757d. *text-align*: *center*; Mengatur teks agar rata tengah.

c. Halaman Profil Admin

Untuk mengakses profil admin dengan mengklik profil pada menu.

Pada halaman ini admin dapat mengubah username, email, dan password sesuai dengan yang diinginkan.



Gambar 4. 46 Halaman *Profil Admin*

```
<div class="container mt-5">
  <h1>Profil Admin</h1>

  <?php if (isset($_SESSION['message'])): ?>
    <div class="alert alert-success">
      <? = htmlspecialchars($_SESSION['message']) ?>
      <?php unset($_SESSION['message']); ?>
    </div>
  <?php endif; ?>

  <form action="admin_profile.php" method="POST">
    <div class="form-group mb-3">
      <label for="username">Username</label>
      <input type="text" class="form-control" id="username" name="username"
        value="<? = htmlspecialchars($admin['username']) ?>" required>
    </div>
    <div class="form-group mb-3">
      <label for="email">Email</label>
      <input type="email" class="form-control" id="email" name="email" value="<? =
        htmlspecialchars($admin['email']) ?>" required>
    </div>
    <div class="form-group mb-3">
      <label for="password">Kata Sandi Baru</label>
      <input type="password" class="form-control" id="password" name="password"
        placeholder="Isi untuk mengganti kata sandi">
    </div>
    <button type="submit" class="btn btn-primary">Perbarui Profil</button>
  </form>
</div>
```

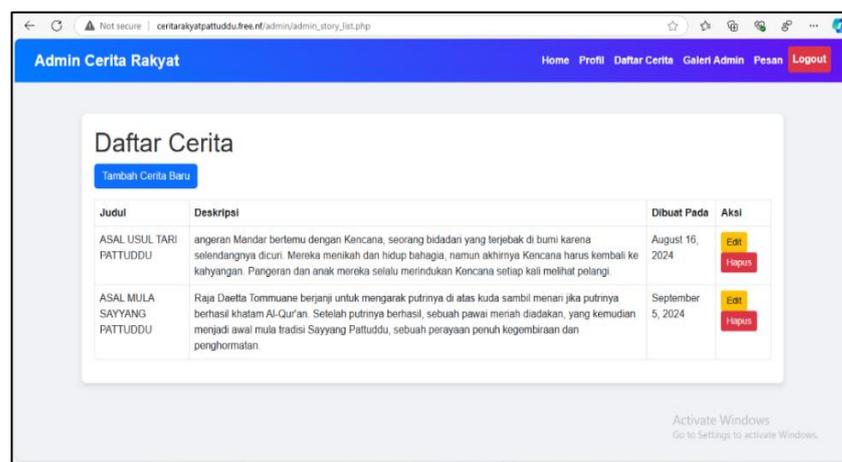
Gambar 4. 47 *Codingan* Halaman *Profil Admin*

Berikut adalah penjelasan bagaimana kode ini bekerja untuk menghasilkan elemen-elemen dalam halaman profile admin: Container dan Judul dimana elemen div dengan *class* container mt-5 digunakan untuk membuat container utama dengan margin atas (*top*) sebesar 5 (mt-5). Elemen `<h1>` menampilkan judul "Profil Admin".

Formulir Profil Admin didalam terdiri dari kode 3 Input yaitu Input untuk Username : menampilkan dan memungkinkan pengeditan username admin yang diambil dari variabel `$admin['username']`. Input untuk Email menampilkan email admin dan memungkinkan pengguna untuk memperbaruinya. Input untuk Kata Sandi Baru ini memungkinkan admin untuk memasukkan kata sandi baru. Jika tidak ingin mengubah kata sandi, kolom ini dapat dibiarkan kosong. Tombol perbarui profil tombol ini mengirimkan formulir ke file `admin_profile.php` untuk diproses.

d. Halaman Daftar Cerita Admin

Untuk mengakses daftar cerita admin dengan mengklik daftar cerita pada menu. Pada halaman ini admin dapat melihat, mengubah, dan menghapus cerita.



Gambar 4. 48 Halaman Daftar Cerita Admin

```

<?php include 'navbar_admin.php'; ?>

<div class="container mt-5">
  <h1 class="page-title">Daftar Cerita</h1>
  <a href="admin_add_story.php" class="btn btn-primary mb-3">Tambah Cerita Baru</a>

  <table class="table table-bordered">
    <thead>
      <tr>
        <th>Judul</th>
        <th>Deskripsi</th>
        <th>Dibuat Pada</th>
        <th>Aksi</th>
      </tr>
    </thead>
    <tbody>
      <?php foreach ($stories as $story): ?>
        <tr>
          <td><?= htmlspecialchars($story['title']) ?></td>
          <td><?= htmlspecialchars($story['description']) ?></td>
          <td><?= date('F j, Y', strtotime($story['created_at'])) ?></td>
          <td>
            <a href="admin_edit_story.php?id=<?= $story['id'] ?>" class="btn btn-warning btn-sm">Edit</a>
            <a href="admin_delete_story.php?id=<?= $story['id'] ?>" class="btn btn-danger btn-sm" onclick="return confirm('Apakah Anda yakin ingin menghapus cerita ini?');">Hapus</a>
          </td>
        </tr>
      <?php endforeach; ?>
    </tbody>
  </table>
</div>
<scriptsrc="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0alpha1/dist/js/bootstrap.bundle.min.js"></script>
</body>

```

Gambar 4. 49 *Codingan* Halaman Daftar Cerita

```

.page-title {
  font-size: 1.5rem;
  font-weight: bold;
  text-align: center;
  color: #007bff;
  margin-bottom: 20px;
}
.btn-warning {
  background-color: #ffc107;
  border-color: #ffc107;
  color: #fff;
}
.btn-warning:hover {
  background-color: #e0a800;
  border-color: #d39e00;
}
.btn-danger {
  background-color: #dc3545;
  border-color: #dc3545;
  color: #fff;
}
.btn-danger:hover {
  background-color: #c82333;
  border-color: #bd2130;
}

```

Gambar 4. 50 CSS Halaman Daftar Cerita

1. Penjelasan Kode Halaman Daftar Cerita

Kode PHP digunakan untuk membuat halaman Daftar Cerita, di mana daftar cerita, dimana data cerita ditarik dari basis data dan ditampilkan dalam bentuk tabel. Menampilkan judul halaman yang sudah diformat dengan kelas CSS `page-title`. Tombol tambah cerita baru dimana `href="admin_add_story.php"`: Mengarahkan pengguna ke halaman untuk menambah cerita baru. `class="btn btn-primary"`: Memberikan gaya tombol biru menggunakan Bootstrap.

Struktur Tabel dimana `table class="table table-bordered"`: Menggunakan gaya tabel *Bootstrap* agar terlihat rapi dengan garis pembatas (*bordered*). **thead**: Bagian untuk menampilkan nama kolom seperti "Judul", "Deskripsi", dll. *Script Bootstrap* memuat fungsi-fungsi *Bootstrap*, seperti tombol dan tabel interaktif.

Menampilkan data cerita menggunakan PHP *Loop* dimana `$stories`: *Array* yang berisi data cerita dari basis data, **foreach**: digunakan untuk menampilkan setiap cerita dalam *table*, **htmlspecialchars**: menghindari injeksi kode dengan mengubah karakter spesial menjadi entitas HTML (misalnya `<` menjadi `<`), **date('F j, Y', strtotime(\$story['created_at']))**: mengubah format tanggal menjadi lebih mudah dibaca. **Edit dan Hapus**: Tombol Edit mengarahkan ke halaman pengeditan cerita berdasarkan Id cerita dan tombol Hapus menampilkan konfirmasi sebelum menghapus data cerita.

CSS digunakan untuk mempercantik tampilan halaman agar lebih menarik, rapi, dan sesuai dengan desain yang diinginkan. CSS untuk Judul Halaman Daftar Cerita menggunakan *font-size*: Mengatur ukuran teks judul agar lebih besar dan menonjol. *font-weight*: Menebalkan teks agar terlihat sebagai elemen penting. *text-*

align: Memusatkan teks. *color*: Memberikan warna biru pada teks judul. *margin-bottom*: Menambahkan ruang kosong di bawah judul agar tidak terlalu rapat dengan tombol tambah cerita baru.

CSS untuk Tabel menggunakan *text-align*: Konten kolom ditampilkan rata tengah secara *horizontal*, membuat tabel lebih rapi. *vertical-align*: Konten ditampilkan di tengah secara vertikal untuk keseimbangan. *font-size*: Menyesuaikan ukuran teks agar tidak terlalu besar.

CSS untuk tombol aksi (Edit dan Hapus) menggunakan *btn-warning*: Digunakan untuk tombol Edit. Warna kuning diatur dengan *hover effect* untuk transisi lebih menarik. *btn-danger*: Digunakan untuk tombol Hapus. Warna merah diatur dengan *hover effect*.

e. Halaman Galeri Admin

Untuk mengakses galeri admin dengan mengklik galeri admin pada menu. Pada halaman ini admin dapat mengupload item baru, mengedit item, dan menghapus item.

The screenshot shows the 'Admin Cerita Rakyat' interface. The 'Galeri Admin' section contains a table with the following data:

ID	Judul	Tipe	Kategori	Aksi
1	TARI PATTUDDU	video	Tari Pattuddu	Edit Hapus
2	SAYYANG PATTUDDU	video	Sayyang Pattuddu	Edit Hapus
3	Kostum yang digunakan oleh Penari Pattuddu	image	Tari Pattuddu	Edit Hapus
4	Musik Iringan Tari Pattuddu	image	Tari Pattuddu	Edit Hapus
5	Festival Sayyang Pattuddu	image	Sayyang Pattuddu	Edit Hapus
6	Sayyang Pattuddu di Desa Dakka, Kec. Tapango	image	Sayyang Pattuddu	Edit Hapus
7	Festival Sayyang Pattuddu di Dinas Kebudayaan dan Parawisata Kabupaten Majene	image	Sayyang Pattuddu	Edit Hapus
8	Gerakan Tari Pattuddu yang ditarikan formasi genap	image	Tari Pattuddu	Edit Hapus

Gambar 4. 51 Halaman Galeri Admin

```

<div class="container mt-4">
  <h1 class="page-title">Galeri Admin</h1>
  <a href="upload_gallery_item.php" class="btn btn-primary mb-4">Upload Item Baru</a>
  <table class="table table-bordered">
    <thead>
      <tr>
        <th>ID</th>
        <th>Judul</th>
        <th>Tipe</th>
        <th>Kategori</th>
        <th>Aksi</th>
      </tr>
    </thead>
    <tbody>
      <?php foreach ($galleryItems as $item): ?>
        <tr>
          <td><?= htmlspecialchars($item['id']) ?></td>
          <td><?= htmlspecialchars($item['title']) ?></td>
          <td><?= htmlspecialchars($item['media_type']) ?></td>
          <td><?= htmlspecialchars($item['category']) ?></td>
          <td>
            <a href="edit_item.php?id=<?= htmlspecialchars($item['id']) ?>"
            class="btn btn-info btn-sm">Edit</a>
            <a href="admin_gallery.php?delete=<?= htmlspecialchars($item['id'])
            ?>" class="btn btn-danger btn-sm" onclick="return confirm('Apakah Anda yakin ingin menghapus
            item ini?');">Hapus</a>
          </td>
        </tr>
      <?php endforeach; ?>
    </tbody>
  </table>
</div>

```

Gambar 4. 53 *Codingan* Halaman Galeri Admin

```

.page-title {
  font-size: 1.5rem;
  font-weight: bold;
  text-align: center;
  color: #007bff;
  margin-bottom: 20px;
}
.btn-warning {
  background-color: #ffc107;
  border-color: #ffc107;
  color: #fff;
}
.btn-warning:hover {
  background-color: #e0a800;
  border-color: #d39e00;
}
.btn-danger {
  background-color: #dc3545;
  border-color: #dc3545;
  color: #fff;
}
.btn-danger:hover {
  background-color: #c82333;
  border-color: #bd2130;
}

```

Gambar 4. 52 *CSS* Halaman Galeri Admin

1. Penjelasan Kode Halaman Galeri Admin

Kode PHP digunakan untuk membuat halaman galeri admin, dimana data ditarik dari basis data dan ditampilkan dalam bentuk tabel. Menampilkan judul halaman yang sudah diformat dengan kelas CSS *page-title*. *Kontainer* (.container): Membuat area utama tempat tabel ditampilkan, elemen h1 dan tombol *Upload* Item Baru ditempatkan di atas tabel.

Tabel Dinamis: *foreach* (\$galleryItems as \$item): iterasi setiap item dari *array* \$galleryItems untuk menampilkan datanya dalam tabel. Data Kolom: `<?= htmlspecialchars($item['id']) ?>`: Menghindari serangan XSS dengan mem-filter *output*. Tombol Aksi (Edit dan Hapus): Tombol Edit: Mengarahkan ke halaman `edit_item.php` dengan id item. Tombol Hapus: `admin_gallery.php?delete=...`: Melakukan penghapusan data. `onclick="return confirm(...)"`: Menambahkan dialog konfirmasi untuk penghapusan.

CSS digunakan untuk mempercantik tampilan halaman agar lebih menarik, rapi, dan sesuai dengan desain yang diinginkan. CSS untuk judul halaman galeri admin menggunakan *font-size*: Mengatur ukuran teks judul agar lebih besar dan menonjol. *font-weight*: Menebalkan teks agar terlihat sebagai elemen penting. *text-align*: Memusatkan teks. *color*: Memberikan warna biru pada teks judul. *margin-bottom*: Menambahkan ruang kosong di bawah judul agar tidak terlalu rapat dengan tombol tambah cerita baru.

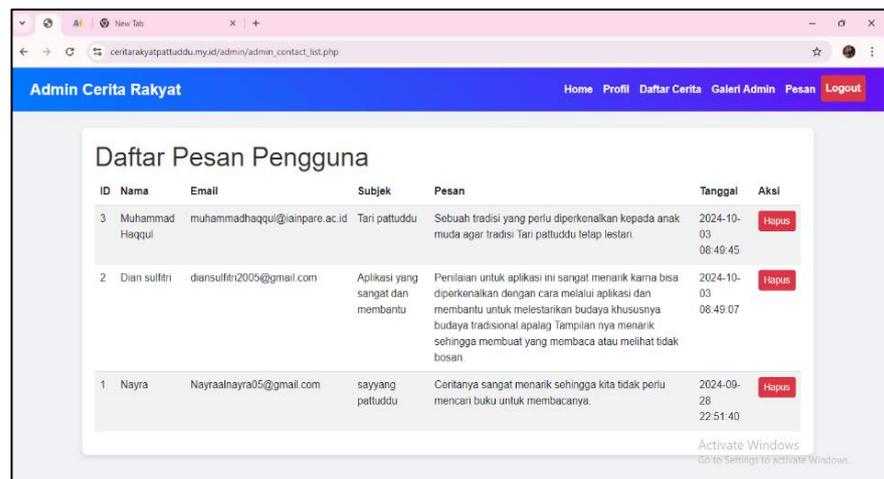
CSS untuk Tabel menggunakan *text-align*: Konten kolom ditampilkan rata tengah secara horizontal, membuat tabel lebih rapi. *vertical-align*: Konten

ditampilkan di tengah secara vertikal untuk keseimbangan. font-size: Menyesuaikan ukuran teks agar tidak terlalu besar.

CSS untuk tombol aksi (Edit dan Hapus) menggunakan btn-warning: Digunakan untuk tombol *Edit*. Warna kuning diatur dengan *hover effect* untuk transisi lebih menarik. btn-danger: Digunakan untuk tombol *Hapus*. Warna merah diatur dengan *hover effect*.

f. Halaman Pesan Admin

Untuk mengakses pesan admin dengan mengklik pesan pada menu. Pada halaman ini admin dapat melihat dan menghapus pesan pengguna.



Gambar 4. 54 Halaman Pesan Admin

```
<div class="container mt-4">
  <h1 class="page-title">Daftar Pesan Pengguna</h1>

  <?php if ($deleteMessage === 'success'): ?>
    <div class="alert alert-success">
      Pesan berhasil dihapus.
    </div>
  <?php endif; ?>

  <div class="table-responsive">
    <table class="table table-striped">
```

```

<thead>
  <tr>
    <th>ID</th>
    <th>Nama</th>
    <th>Email</th>
    <th>Subjek</th>
    <th>Pesan</th>
    <th>Tanggal</th>
    <th>Aksi</th>
  </tr>
</thead>
<tbody>
  <?php if (!empty($comments)): ?>
    <?php foreach ($comments as $comment): ?>
      <tr>
        <td><?php echo htmlspecialchars($comment['id']); ?></td>
        <td><?php echo htmlspecialchars($comment['name']); ?></td>
        <td><?php echo htmlspecialchars($comment['email']); ?></td>
        <td><?php echo htmlspecialchars($comment['subject']); ?></td>
        <td><?php echo htmlspecialchars($comment['message']); ?></td>
        <td><?php echo htmlspecialchars($comment['created_at']); ?></td>
        <td>
          <a href="delete_message.php?id=<?php echo $comment['id']; ?>"
            class="btn btn-danger btn-sm" onclick="return confirm('Apakah
            Anda yakin ingin menghapus pesan ini?');">Hapus</a>
        </td>
      </tr>
    <?php endforeach; ?>
  <?php else: ?>
    <tr>
      <td colspan="7">Tidak ada pesan yang ditemukan.</td>
    </tr>
  <?php endif; ?>
</tbody>
</table>
</div>
</div>

```

Gambar 4. 56 *Codingan* Halaman Pesan Pengguna

```

.page-title {
  font-size: 1.5rem;
  font-weight: bold;
  text-align: center;
  color: #007bff;
  margin-bottom: 20px;
}
.btn-danger {
  background-color: #dc3545;
  border-color: #dc3545;
  color: #fff;
}
.btn-danger:hover {
  background-color: #c82333;
  border-color: #bd2130;
}

```

Gambar 4. 55 *CSS* Halaman Pesan Pengguna

1. Penjelasan Kode Halaman Galeri Admin

Kode PHP digunakan untuk membuat halaman pesan pengguna, dimana data ditarik dari basis data dan ditampilkan dalam bentuk tabel. Menampilkan judul

halaman yang sudah diformat dengan kelas CSS *page-title*. `<div class="table-responsive">`: Membuat tabel dapat di-*scroll* secara *horizontal* jika jumlah kolom terlalu banyak (khusus untuk layar kecil). `<table class="table table-striped">`: *table-class Bootstrap* untuk memberikan gaya default pada tabel. *table-striped*: Menambahkan efek bergaris pada baris genap untuk mempermudah membaca data.

Header Tabel Header kolom seperti ID, Nama, *Email*, Subjek, Pesan, Tanggal, Aksi merepresentasikan data pengguna yang akan ditampilkan. `<thead>`: Bagian untuk mendefinisikan header tabel. `<tr>`: Baris pertama tabel, digunakan untuk nama kolom. `<th>`: Elemen tabel untuk mendefinisikan *header* kolom.

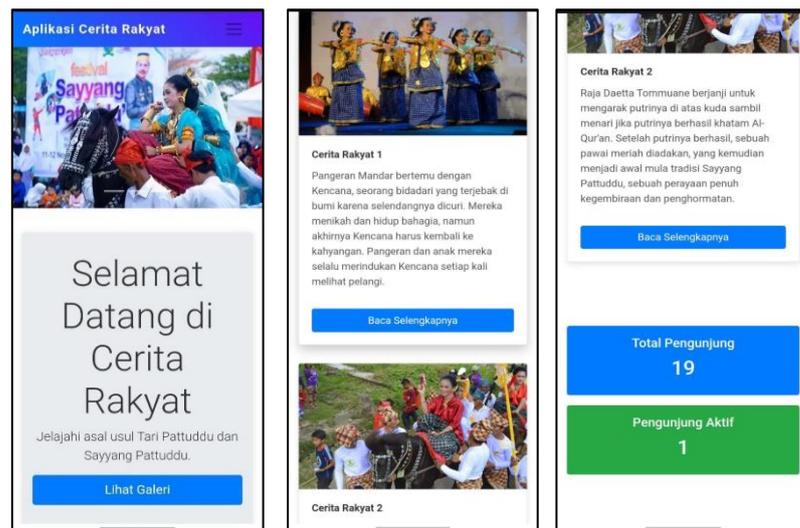
Baris Data `foreach ($comments as $comment)`: Loop ini digunakan untuk iterasi setiap data pesan dalam array `$comments`. Setiap elemen dalam array berisi informasi pesan pengguna. `htmlspecialchars()`: digunakan untuk mencegah XSS (Cross-Site Scripting) dengan mengonversi karakter khusus seperti `<`, `>` menjadi entitas HTML aman. `<tr>` dan `<td>`: `<tr>`: Setiap baris data dalam table dan `<td>`: Setiap sel dalam baris, digunakan untuk menampilkan data seperti id dan lainnya.

Tombol (Hapus) `href="delete_message.php?id=<?php echo $comment['id'] ; ?>"`: URL untuk menghapus pesan. Menggunakan id unik dari pesan untuk menentukan pesan mana yang akan dihapus. `class="btn btn-danger btn-sm"`: *btn btn-danger*: Tombol gaya Bootstrap dengan warna merah. *btn-sm*: Menjadikan tombol lebih kecil. `onclick="return confirm(...)"`: Membuka dialog konfirmasi sebelum tombol dieksekusi. Jika pengguna menekan *OK*, maka tindakan dihapus dilanjutkan. Div terakhir menutup elemen pembungkus untuk memastikan struktur halaman tetap rapi.

2. User

a. Halaman *Dashboard*

Halaman ini merupakan *dashboard* pengguna. Menampilkan yang berisi ringkasan informasi, akses cepat untuk lihat galeri, pengguna dapat langsung memilih cerita yang diinginkan dan pengguna dapat melihat total pengguna dan pengguna aktif yang mengunjungi aplikasi.



Gambar 4. 57 Halaman Dashboard Pengguna

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Home Page</title>
  <link rel="stylesheet" href="css/styles.css">
  <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>
  <?php include 'navbar.php'; ?>
  <div id="carouselExampleIndicators" class="carousel slide" data-ride="carousel">
    <ol class="carousel-indicators">
      <li data-target="#carouselExampleIndicators" data-slide-to="0" class="active"></li>
      <li data-target="#carouselExampleIndicators" data-slide-to="1"></li>
      <li data-target="#carouselExampleIndicators" data-slide-to="2"></li>
    </ol>
    <div class="carousel-inner">
      <div class="carousel-item active">
        
        <div class="carousel-caption d-none d-md-block">
          <h5>Tari Pattuddu</h5>
          <p>Tarian tradisional dari tanah mandar memiliki gerakan lemah gemulai</p>
        </div>
      </div>
      <div class="carousel-item">
```

```

        
        <div class="carousel-caption d-none d-md-block">
            <h5>Parrawana yang memainkan rebana</h5>
            <p>Rebana adalah Musik Iringan Sayyang Pattuddu yang biasa diselingi kalindaqda (Puisi
            khas Mandar) </p>
        </div>
    </div>
    <div class="carousel-item">
        
        <div class="carousel-caption d-none d-md-block">
            <h5>Festival Sayyang Pattuddu</h5>
            <p>Lapangan Bala, Pampusuang. Acara menampilkan keindahan tradisi arak-arak dan pertunjukan
            Sayyang pattuddu yang melibatkan ratusan kuda.</p>
        </div>
    </div>
    </div>
    <a class="carousel-control-prev" href="#carouselExampleIndicators" role="button" data-slide="prev">
        <span class="carousel-control-prev-icon" aria-hidden="true"></span>
        <span class="sr-only">Previous</span>
    </a>
    <a class="carousel-control-next" href="#carouselExampleIndicators" role="button" data-slide="next">
        <span class="carousel-control-next-icon" aria-hidden="true"></span>
        <span class="sr-only">Next</span>
    </a> </div>
<div class="container mt-4">
<div class="jumbotron text-center">
    <h1 class="display-4">Selamat Datang di Cerita Rakyat</h1>
    <p class="lead">Jelajahi asal usul Tari Pattuddu dan Sayyang Pattuddu.</p>
    <a class="btn btn-primary btn-lg" href="gallery.php" role="button">Lihat Galeri</a>
</div>
<div class="row">
    <div class="col-lg-4 col-md-6 mb-4">
        <div class="card">
            
            <div class="card-body">
                <h5 class="card-title">Cerita Rakyat 1</h5>
                <p class="card-text">Pangeran Mandar bertemu dengan Kencana, seorang bidadari yang terjebak
                di bumi karena selendangnya dicuri. Mereka menikah dan hidup bahagia, namun akhirnya Kencana
                harus kembali ke kayangan. Pangeran dan anak mereka selalu merindukan Kencana setiap kali
                melihat pelangi.</p>
                <a href="story_detail.php?id=1" class="btn btn-primary">Baca Selengkapnya</a>
            </div>
        </div>
    </div>
    <div class="col-lg-4 col-md-6 mb-4">
        <div class="card">
            
            <div class="card-body">
                <h5 class="card-title">Cerita Rakyat 2</h5>
                <p class="card-text">Raja Daetta Tommuane berjanji untuk mengarak putrinya di atas kuda
                sambil menari jika putrinya berhasil khatam Al-Qur'an. Setelah putrinya berhasil, sebuah
                pawai meriah diadakan, yang kemudian menjadi awal mula tradisi Sayyang Pattuddu, sebuah
                perayaan penuh kegembiraan dan penghormatan.</p>
                <a href="story_detail.php?id=2" class="btn btn-primary">Baca Selengkapnya</a>
            </div>
        </div>
    </div>
</div>
</div>
</div>
<div class="container my-5">
    <div class="row justify-content-center">
        <div class="col-lg-3 col-md-4 col-sm-6 mb-3">
            <div class="stat-card bg-primary text-white text-center p-3 rounded">
                <h5 class="stat-title">Total Pengunjung</h5>
                <p class="stat-number h2 fw-bold"><?= htmlspecialchars($total_visitors); ?></p>
            </div>
        </div>
        <div class="col-lg-3 col-md-4 col-sm-6 mb-3">
            <div class="stat-card bg-success text-white text-center p-3 rounded">
                <h5 class="stat-title">Pengunjung Aktif</h5>
                <p class="stat-number h2 fw-bold"><?= htmlspecialchars($active_visitors); ?></p>
            </div>
        </div>
    </div>
</div>
</div>

```

Gambar 4. 58 Codingan Halaman Dashboard User

```

.navbar {
  background: linear-gradient(45deg, #007bff, #6610f2);
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
  padding: 10px 20px;
  z-index: 1000;
  position: sticky;
  top: 0;
  width: 100%;
}
.navbar-toggler{
  padding: .25rem .75rem;
  font-size: 1.25rem;
  line-height: 1;
  background-color: transparent;
  border: 1px solid transparent;
  border-radius: .25rem;
}
.jumbotron {
  background-color: #ffffff;
  background-image: url('your-image-path.jpg');
  background-size: cover;
  background-position: center;
  padding: 2rem 1rem;
  border-radius: 8px;
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
  text-align: center;
  color: #333;
}
.card {
  display: flex;
  flex-direction: column;
  height: 100%;
  border-radius: 8px;
  overflow: hidden;
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
  transition: transform 0.2s ease, box-shadow 0.2s ease;
}

```

Gambar 4. 59 CSS Halaman *Dashboard User*

1. Penjelasan Kode Halaman *Dashboard User*

Pada gambar 4.53 tersebut adalah kode HTML yang menggunakan PHP dan Bootstrap untuk membuat halaman web. *Header HTML* yang terdiri dari `<!DOCTYPE html>` untuk mendeklarasikan dokumen sebagai HTML5. `<html lang="en">` untuk mendefinisikan bahasa dokumen sebagai Inggris. `<head>` berisi metadata dan link ke file eksternal seperti `meta charset="UTF-8"` untuk mengatur encoding karakter ke UTF-8 dan `meta name="viewport"`: Membuat halaman responsif terhadap berbagai ukuran layar. Title untuk menentukan judul halaman (ditampilkan di tab browser) dan link untuk menghubungkan ke file CSS lokal (styles.css) dan Bootstrap CDN untuk gaya.

Kode `<?php include 'navbar.php'; ?>` untuk memasukkan file PHP eksternal untuk navbar. Kode *Carousel* bagian slider yang menampilkan beberapa gambar secara bergantian terdiri dari *carousel-indicators*: Tombol untuk mengganti slide secara manual. *carousel-inner*: Berisi tiga slide dengan deskripsi dan gambar terkait tradisi Mandar (Tari Pattuddu, musik rebana, dan Festival Sayyang Pattuddu). *carousel-control-prev* dan *carousel-control-next*: Navigasi untuk mengganti slide sebelumnya dan berikutnya.

container mt-4: Bagian utama konten, berisi *Jumbotron* untuk menampilkan pesan sambutan, deskripsi singkat, dan tombol untuk melihat galeri. *Card Layout* yang terdiri tiga kolom responsif menggunakan *Bootstrap grid*: *Card 1* untuk Cerita rakyat tentang Pangeran Mandar. *Card 2* untuk Cerita tentang tradisi Sayyang Pattuddu. Tombol "Baca Selengkapnya" membawa pengguna ke detail cerita melalui `story_detail.php?id`.

Statistik Pengunjung dengan kode `<?= htmlspecialchars($total_visitors); ?>` untuk menampilkan jumlah total pengunjung, dan `<?= htmlspecialchars($active_visitors); ?>` untuk menampilkan jumlah pengunjung aktif. Bootstrap digunakan untuk membuat desain kartu statistik yang menarik.

Script Menambahkan fungsionalitas dengan pustaka eksternal: *jQuery*: Untuk memanipulasi DOM dan mendukung *Bootstrap JS*. *Popper.js*: Mengelola posisi elemen (misalnya *dropdown*, *tooltip*). *Bootstrap.min.js*: Menyediakan fitur interaktif *Bootstrap*, seperti *carousel*. PHP digunakan untuk memuat elemen dinamis seperti `navbar.php`, total pengunjung (`$total_visitors`), dan pengunjung

aktif (`$active_visitors`).Bootstrap memastikan tampilan yang baik di berbagai perangkat (desktop, tablet, atau ponsel).

`.navbar` berfungsi untuk mengatur tampilan navbar. Rincian kodenya: *background*: `linear-gradient(45deg, #007bff, #6610f2)`; untuk memberikan latar belakang gradasi biru ke ungu. *box-shadow* untuk menambahkan bayangan untuk memberi efek kedalaman. *Padding* untuk menambahkan ruang di dalam navbar (vertikal 10px, horizontal 20px). *z-index*: 1000 memastikan navbar tetap di atas elemen lain. *position*: `sticky` untuk membuat navbar tetap di atas saat halaman digulir. *top*: 0 memastikan navbar tetap menempel di bagian atas. *width*: 100% memastikan navbar melebar penuh.

`.navbar-toggler` berfungsi menyesuaikan tombol navigasi untuk tampilan kecil. *padding*: Mengatur ruang di dalam tombol. *font-size*: Mengatur ukuran teks atau ikon pada tombol. *line-height*: Menentukan tinggi baris. *background-color*: Menjadikan tombol transparan. *border*: Mengatur garis batas tombol agar transparan. *border-radius*: Membulatkan sudut tombol.

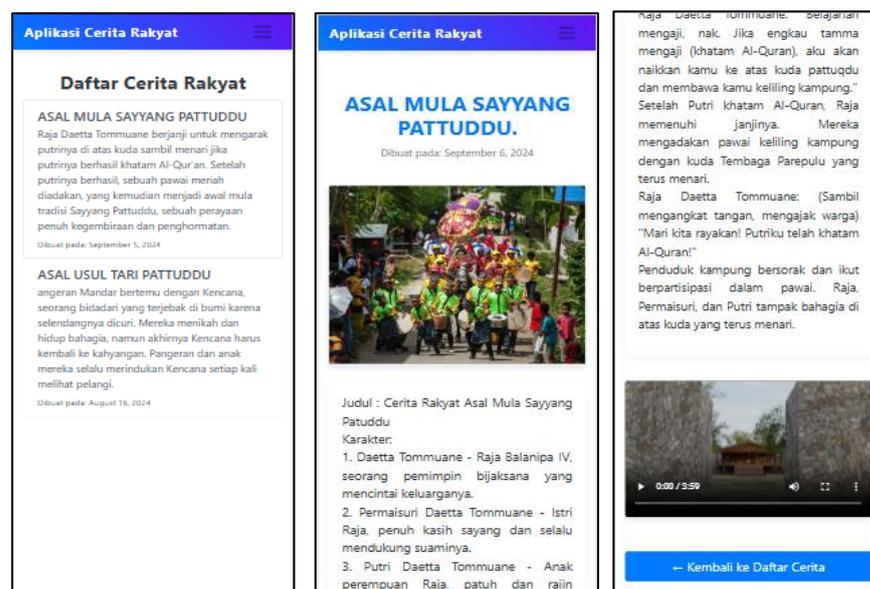
`.jumbotron` berfungsi menyesuaikan tampilan *hero section* di bagian atas halaman. *background-color*: `#ffffff`: Memberikan warna latar belakang putih. *background-image*: Menambahkan gambar sebagai latar belakang. *background-size*: `cover`: Membuat gambar latar menyesuaikan ukuran elemen. *background-position*: `center`: Memusatkan posisi gambar. *padding*: Menambahkan ruang di sekitar konten.

`.card` berfungsi mengatur gaya elemen kartu yang digunakan dalam *grid*. *display*: `flex`; *flex-direction*: `column`; Membuat tata letak kartu fleksibel secara

vertikal. *height: 100%;*; Memastikan kartu menyesuaikan ketinggian. *border-radius:* Membulatkan sudut kartu. *overflow: hidden:* Menyembunyikan elemen yang keluar dari batas kartu. *box-shadow:* Menambahkan bayangan untuk memberikan kesan melayang. *transition:* Memberikan animasi halus saat kartu diinteraksi: *transform:* Untuk efek perubahan posisi atau skala. *box-shadow:* Untuk perubahan efek bayangan saat *hover*. *border-radius:* Membulatkan sudut elemen. *box-shadow:* Menambahkan bayangan untuk memberi efek melayang. *text-align: center:* Memusatkan teks secara *horizontal*. *color: #333:* Memberikan warna teks abu-abu gelap untuk kontras.

b. Halaman Cerita

Pengguna memilih menu cerita. Menampilkan daftar cerita yang tersedia berisi judul dan deskripsi singkat. Pengguna memilih cerita yang ingin dibaca atau ditonton. Menampilkan detail cerita yang dipilih pengguna dapat membaca dan menonton cerita



Gambar 4. 60 Halaman Cerita

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Daftar Cerita Rakyat</title>
  <link rel="stylesheet" href="css/styles.css">
  <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
    rel="stylesheet">
</head>
<body>
  <?php include 'navbar.php'; ?>

  <div class="container mt-5">
    <h1 class="page-title">Daftar Cerita Rakyat</h1>
    <div class="list-group">
      <?php foreach ($stories as $story): ?>
        <a href="story_detail.php?id=<?=$story['id'] ?>" class="list-group-item list-
          group-item-action">
          <h5 class="mb-1"><?=$story['title'] ?></h5>
          <p class="mb-1"><?=$story['description'] ?></p>
          <small>Dibuat pada: <?=$story['created_at'] ?></small>
        </a>
      <?php endforeach; ?>
    </div>
  </div>

  /*Story Detail*/
  <div class="container mt-5">
    <div class="story-header">
      <h1 class="story-title"><?=$story['title'] ?></h1>
      <p class="story-meta">Dibuat pada: <?=$story
        ['created_at'] ?> </p>
    </div>
    <?php if (!empty($story['image'])): ?>
      <div class="story-image mb-4">
        
      </div>
    <?php endif; ?>
    <div class="story-content">
      <p><?=$story['content'] ?></p>
    </div>
    <?php if (!empty($story['video'])): ?>
      <div class="story-video mt-4">
        <video controls class="w-100 rounded">
          <source src="videos/<?=$story['video'] ?>" type="video/mp4">
          Browser Anda tidak mendukung pemutaran video.
        </video>
      </div>
    <?php endif; ?>
    <div class="mt-5 text-center">
      <a href="story_list.php" class="btn btn-primary btn-lg">&larr; Kembali ke Daftar
        Cerita</a>
    </div>
  </div>
  <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
  <script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.9.2/dist/umd/popper.min.js">
</script>
  <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js">
</script>
</body>
</html>

```

Gambar 4. 61 *Codingan* Halaman Cerita

```

.story-header {
  text-align: center;
  margin-bottom: 40px;
}
.story-title {
  font-size: 2.5rem;
  color: #007bff;
  margin-bottom: 10px;
  font-weight: bold;
}
.story-meta {
  font-size: 1rem;
  color: #6c757d;
}
.story-image img {
  width: 100%;
  height: auto;
  border-radius: 15px;
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
}
.story-content {
  font-size: 1.2rem;
  color: #343a40;
  margin-bottom: 40px;
  text-align: justify;
  background: #ffffff;
  padding: 20px;
  border-radius: 10px;
  box-shadow: 0 4px 12px rgba(0, 0, 0, 0.05);
}
.story-video video {
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
}

```

Gambar 4. 62 CSS Halaman Cerita

2. Penjelasan Kode Halaman Daftar Cerita

Kode HTML `<!DOCTYPE html>` untuk mendeklarasikan dokumen sebagai HTML5. `<html lang="en">`: bahasa utama dokumen adalah Inggris. `<head>`: mengatur metadata halaman: *encoding* UTF-8, *viewport* responsif, judul halaman dan menghubungkan file CSS lokal (*styles.css*) dan framework *Bootstrap* untuk gaya. Menggunakan navbar `<?php include 'navbar.php'; ?>` untuk memuat file PHP eksternal berisi navigasi.

Pada halaman daftar cerita kontainer utama: `class="container mt-5"`: Membuat *margin* atas 5 unit dengan *Bootstrap*. `h1.page-title`: menampilkan judul halaman "Daftar Cerita Rakyat". *List Group*: menggunakan *foreach* PHP untuk menampilkan daftar cerita dari array `$stories`: Link ke Detail:

href="story_detail.php?id=<?=\$story['id'] ?>" mengarah ke halaman detail cerita dengan Id spesifik. menampilkan: Judul: \$story['title']. Deskripsi singkat: \$story['description']. Tanggal Pembuatan: format tanggal menggunakan fungsi *date()*.

Halaman detail cerita terdiri dari judul untuk menampilkan judul cerita (\$story['title']) dan tanggal dibuat (\$story['created_at']). Gambar Cerita (Opsional): Jika \$story['image'] tidak kosong, gambar akan ditampilkan dengan kelas *img-fluid* (responsif). Konten Cerita: menggunakan *nl2br()* untuk menambahkan baris baru, dan *htmlspecialchars()* untuk keamanan (mencegah injeksi). Video Cerita (Opsional): Jika \$story['video'] tersedia, ditampilkan dengan tag `<video>` responsif. Tombol navigasi kembali ke daftar: tombol mengarah ke halaman daftar cerita dengan link *story_list.php*.

Kode CSS ini bertujuan memberikan tampilan profesional dan nyaman untuk halaman cerita. Penjelasan Kode CSS *.story-header* untuk tampilan *Header* Cerita: *text-align: center*: memusatkan teks di bagian *header*. *margin-bottom: 40px*: memberikan jarak bawah 40px. *.story-title* untuk judul cerita: *font-size: 2.5rem*: ukuran teks besar untuk menonjolkan judul. *color: #007bff*: warna biru untuk menarik perhatian. *margin-bottom: 10px*: jarak bawah untuk memisahkan judul dari elemen lainnya. *font-weight: bold*: menjadikan teks tebal.

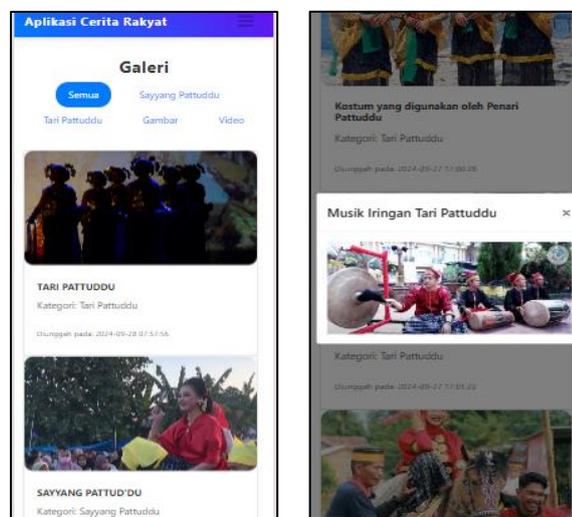
.story-meta untuk metadata cerita *font-size: 1rem*: Ukuran teks kecil. *color: #6c757d*: Warna abu-abu untuk menandakan informasi tambahan. *.story-image* *img* gambar cerita: *width: 100%*: gambar mengisi seluruh lebar elemen induk. *height*:

auto: proporsi gambar tetap sesuai aslinya. *border-radius*: 15px: membulatkan sudut gambar. *box-shadow*: menambahkan bayangan lembut untuk efek kedalaman.

.story-content untuk konten cerita: *font-size*: 1.2rem: ukuran teks sedang agar nyaman dibaca. *color*: #343a40: warna teks abu-abu gelap. *margin-bottom*: 40px: memberikan jarak bawah dengan elemen berikutnya. *text-align*: *justify*: menyalarkan teks kiri-kanan untuk tata letak rapi. *background*: #ffffff: latar belakang putih untuk kontras. *padding*: 20px: memberikan ruang di dalam elemen. *border-radius*: 10px: membulatkan sudut. *box-shadow*: bayangan lembut untuk efek melayang. *.story-video* untuk video cerita: *box-shadow*: Menambahkan bayangan lembut untuk memberi fokus pada video.

c. Halaman Galeri

Pengguna memilih menu galeri. Menampilkan isi galeri yang berisi deskripsi singkat, gambar dan video yang terkait dengan cerita. Pengguna dapat melihat gambar lebih detail, memutar dan mendengar penjelasan dalam video yang ada.



Gambar 4. 63 Halaman Galeri

```

<?php include 'navbar.php'; ?>

<div class="container mt-5">
  <h1 class="page-title text-center">Galeri</h1>
  <div class="category-navbar text-center">
    <?php foreach ($categories as $key => $category): ?>
      <a href="gallery.php?category=<?=> htmlspecialchars($key) ?>" class="<?=> $selectedCategory ==
        $key ? 'active' : '' ?>">
        <?=> htmlspecialchars($category) ?> </a>
    <?php endforeach; ?>
  </div>
</div>
<div class="row mt-4">
  <?php if ($galleryItems): ?>
    <?php foreach ($galleryItems as $item): ?>
      <div class="col-md-4">
        <div class="card gallery-item shadow-sm">
          <?php if ($item['media_type'] == 'image'): ?>
            <?php if (file_exists('uploads/' . $item['file_path'])) && !empty ($item['file_path']): ?>
              <a href="#" class="media-link" data-toggle="modal" data-target="#mediaModal" data-
                media="uploads/<?=> htmlspecialchars($item['file_path']) ?>" data-type="image"
                data-title="<?=> htmlspecialchars($item['title']) ?>">
               htmlspecialchars($item['title']) ?>">
            </a>
          <?php else: ?>
            
          <?php endif; ?>
          <?php elseif ($item['media_type'] == 'video'): ?>
            <?php if (file_exists('uploads/' . $item['file_path']) && !empty($item ['file_path'])): ?>
              <a href="#" class="media-link" data-toggle="modal" data-target="#mediaModal" data-
                media="uploads/<?=> htmlspecialchars($item['file_path']) ?>" data-type="video"
                data-title="<?=> htmlspecialchars($item['title']) ?>">
              <video class="card-img-top" muted>
                <source src="uploads/<?=> htmlspecialchars($item['file_path']) ?>" type="video/mp4">
                Browser Anda tidak mendukung tag video.
              </video>
            </a>
          <?php else: ?>
            
          <?php endif; ?>
        <?php endif; ?>
        <div class="card-body">
          <h5 class="card-title"><?=> htmlspecialchars($item['title']) ?></h5>
          <p class="card-text">Kategori: <?=> htmlspecialchars($item['category']) ?></p>
          <p class="card-text"><small class="text-muted">Diunggah pada: <?=> htmlspecialchars(
            $item['created_at']) ?></small></p>
        </div>
      </div>
    </div>
  <?php endforeach; ?>
  <?php else: ?>
    <div class="col-12">
      <p class="text-center">Tidak ada item yang ditemukan untuk kategori ini.</p>
    </div>
  <?php endif; ?>
</div>
</div>
<div class="modal fade" id="mediaModal" tabindex="-1" role="dialog" aria-labelledby="mediaModalLabel"
  aria-hidden="true">
  <div class="modal-dialog modal-dialog-centered" role="document">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title" id="mediaModalLabel">Media Viewer</h5>
        <button type="button" class="close" data-dismiss="modal" aria-label="Close">
          <span aria-hidden="true">&times;</span>
        </button>
      </div>
      <div class="modal-body">
      </div>
    </div>
  </div>
</div>
</div>

```

Gambar 4. 64 Codingan Halaman Galeri

```

.container {
  margin-top: 50px;
  padding: 15px;
}
.gallery-item {
  position: relative;
  border-radius: 15px;
  overflow: hidden;
  margin-bottom: 20px;
  margin-right: 10px;
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
  background-color: #ffffff;
  transition: transform 0.2s ease, box-shadow 0.2s ease;
}
.gallery-item:hover {
  transform: translateY(-5px);
  box-shadow: 0 8px 16px rgba(0, 0, 0, 0.2);
}
.gallery-item img, .gallery-item video {
  width: 100%;
  height: auto;
  border-radius: 15px;
}
.row {
  margin-left: -5px;
  margin-right: -5px;
}
.col-md-4 {
  padding-left: 5px;
  padding-right: 5px;
}

```

Gambar 4. 65 CSS Halaman Galeri

1. Penjelasan Kode Halaman Galeri

Kode ini efektif untuk menampilkan galeri dengan kategori, gambar, video, dan *fallback* (*placeholder*) saat file media tidak ditemukan. Navbar (*Include File*) `<?php include 'navbar.php'; ?>`: menyisipkan file navbar.php untuk menampilkan menu navigasi. *Header* Galeri `h1.page-title`: Menampilkan judul halaman "Galeri" di tengah layar. Navbar kategori untuk menampilkan daftar kategori dengan tautan. Menggunakan *foreach* untuk membuat tautan dari `$categories`. Kategori yang aktif diberi kelas *active* menggunakan perbandingan `$selectedCategory == $key`.

Grid galeri untuk menampilkan item galeri dalam tata letak grid menggunakan *Bootstrap*. *if* (`$galleryItems`): Mengecek apakah ada item dalam `$galleryItems`. Setiap item gambar: mengecek keberadaan file gambar, menampilkan gambar (``), atau *placeholder* jika gambar tidak ditemukan.

gambar dapat dibuka dalam modal. Video: mengecek keberadaan file video, menampilkan video (<video>), atau *placeholder* jika video tidak ditemukan, video dapat dibuka dalam modal. Keterangan: Judul ($\$item['title']$), kategori ($\$item['category']$), dan tanggal unggah ($\$item['created_at']$). Jika tidak ada item: Pesan "Tidak ada item yang ditemukan" ditampilkan.

Modal Media berfungsi: Menampilkan media (gambar/video) dalam tampilan *pop-up*. Konten Modal: *Header* modal: Menampilkan judul media dan *body* modal: Tempat konten media dimuat secara dinamis.

kode CSS ini dengan tampilan gambar galeri Berikut penjelasan lengkap tapi singkat mengenai kode CSS tersebut: *.container* berfungsi: Mengatur jarak dan tata letak utama galeri. Kodenya yaitu *margin-top: 50px;* dan *padding: 15px;* Memberikan ruang di dalam container untuk konten.

.gallery-item berfungsi untuk mengatur gaya untuk setiap item galeri. *position: relative;* Memungkinkan kontrol posisi elemen anak, jika diperlukan, *border-radius: 15px;* Membuat sudut elemen menjadi melengkung, *overflow: hidden;* Menyembunyikan bagian elemen anak yang keluar dari batas item, *margin-bottom: 20px; margin-right: 10px;* Memberikan jarak antar item secara vertikal dan *horizontal*, *box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);* Menambahkan bayangan lembut untuk memberikan efek kedalaman, *background-color: #ffffff;* Memberikan warna latar putih, *transition: transform 0.2s ease, box-shadow 0.2s ease;* Menambahkan efek animasi saat *hover*.

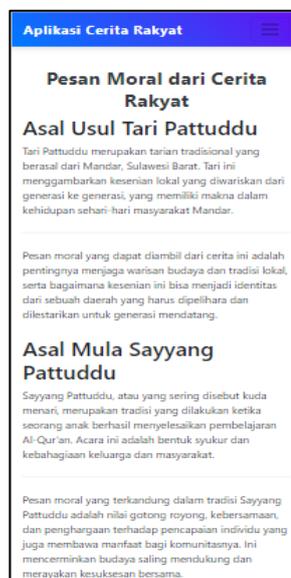
.gallery-item:hover berfungsi untuk menambahkan efek *hover*. *transform: translateY(-5px);* Menggeser elemen sedikit ke atas saat di-*hover*. *box-shadow: 0 8px 16px rgba(0, 0, 0, 0.2);* Memperbesar bayangan untuk memberi efek "mengambang".

`.gallery-item img`, `.gallery-item video` berfungsi untuk mengatur tampilan gambar atau video dalam item galeri. `width: 100%`; Memastikan gambar atau video menyesuaikan lebar item. `height: auto`; Menjaga proporsi gambar/video. `border-radius: 15px`; Membuat sudut media melengkung, menyelaraskan dengan item galeri.

`.row` berfungsi untuk menyesuaikan tata letak grid. `margin-left: -5px`; `margin-right: -5px`; Menetralkan `padding` bawaan dari kolom *Bootstrap*, memastikan item sejajar. `.col-md-4` berfungsi untuk mengatur jarak antar kolom di dalam *grid*. `padding-left: 5px`; `padding-right: 5px`; Memberikan ruang antar item dalam kolom.

d. Halaman Pesan Moral

Pengguna memilih menu pesan moral. Menampilkan pesan moral yang telah ada dan relevan dengan cerita. Pengguna dapat membaca pesan moral yang disajikan dalam bentuk teks.



Gambar 4. 66 Halaman Pesan Moral

```

<?php include 'navbar.php'; ?>
<div class="container mt-5">
  <h1 class="page-title">Pesan Moral dari Cerita Rakyat</h1>
  <article class="mb-4">
    <h2>Asal Usul Tari Pattuddu</h2>
    <p>Tari Pattuddu merupakan tarian tradisional yang berasal dari Mandar, Sulawesi Barat. Tari ini menggambarkan kesenian lokal yang diwariskan dari generasi ke generasi, yang memiliki makna dalam kehidupan sehari-hari masyarakat Mandar.</p>
    <hr class="my-4">
    <p>Pesan moral yang dapat diambil dari cerita ini adalah pentingnya menjaga warisan budaya dan tradisi lokal, serta bagaimana kesenian ini bisa menjadi identitas dari sebuah daerah yang harus dipelihara dan dilestarikan untuk generasi mendatang.</p>
  </article>
  <article>
    <h2>Asal Mula Sayyang Pattuddu</h2>
    <p>Sayyang Pattuddu, atau yang sering disebut kuda menari, merupakan tradisi yang dilakukan ketika seorang anak berhasil menyelesaikan pembelajaran Al-Qur'an. Acara ini adalah bentuk syukur dan kebahagiaan keluarga dan masyarakat.</p>
    <hr class="my-4">
    <p>Pesan moral yang terkandung dalam tradisi Sayyang Pattuddu adalah nilai gotong royong, kebersamaan, dan penghargaan terhadap pencapaian individu yang juga membawa manfaat bagi komunitasnya. Ini mencerminkan budaya saling mendukung dan merayakan kesuksesan bersama.</p>
  </article>
</div>
</div>

```

Gambar 4. 67 *Codingan* Halaman Pesan Moral

```

.container {
  margin-top: 50px; /* Memberikan jarak atas antara navbar dan konten */
  padding: 15px;} /* Menambahkan padding untuk memberikan ruang di sekitar konten */
.page-title {
  font-size: 2rem; /* Ukuran font yang cukup besar untuk menjadi fokus utama */
  text-align: center; /* Memusatkan teks pada tengah halaman */
  margin-bottom: 30px; /* Memberikan jarak bawah agar tidak terlalu rapat dengan konten */
  font-weight: bold; /* Membuat teks terlihat lebih tegas */
  color: #343a40;} /* Warna gelap untuk meningkatkan keterbacaan */
article {
  margin-bottom: 20px; /* Jarak antar artikel untuk tampilan yang lebih terpisah */
  padding: 15px; /* Memberikan ruang di dalam artikel */
  border-radius: 10px; /* Membuat sudut artikel melengkung agar terlihat modern */
  background-color: #ffffff; /* Latar belakang putih untuk kontras dengan teks */
  box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1); } /* Menambahkan bayangan efek floating */
article h2 {
  font-size: 1.5rem; /* Ukuran heading artikel yang cukup besar */
  margin-bottom: 15px; /* Memberikan jarak bawah dengan paragraf berikutnya */
  color: #007bff; /* Warna biru untuk menarik perhatian pada judul */
  font-weight: bold;} /* Membuat teks terlihat tegas */
article p {
  font-size: 1rem; /* Ukuran font standar untuk keterbacaan */
  line-height: 1.6; /* Memberikan ruang antar baris agar lebih nyaman dibaca */
  color: #495057; /* Warna abu-abu gelap untuk kontras dengan latar belakang */
  text-align: justify; /* Meratakan teks untuk estetika yang lebih rapi */
  margin-bottom: 10px;} /* Memberikan jarak antar paragraf */
hr.my-4 {
  border: 0; /* Menghilangkan default border */
  height: 1px; /* Tinggi garis */
  background-color: #e9ecef; /* Warna abu-abu terang untuk garis */
  margin: 20px 0; /* Jarak atas dan bawah agar terlihat nyaman */
}

```

Gambar 4. 68 CSS Halaman Pesan Moral

1. Penjelasan Kode Halaman Pesan Moral

Kode di atas digunakan untuk membuat halaman web yang menampilkan informasi tentang Pesan Moral dari Cerita Rakyat. Berikut penjelasan secara singkat dan detail mengenai kode di halaman ini: navbar `<?php include 'navbar.php'; ?>` berfungsi untuk menyertakan file PHP navbar.php untuk menampilkan navigasi di bagian atas halaman.

Kode kontainer utama : `<div class="container mt-5">` berfungsi membuat area utama halaman dengan *margin* atas (mt-5) agar ada jarak antara navbar dan konten. Kelas container milik *Bootstrap* menjaga tata letak responsif. Kode judul halaman `<h1 class="page-title"> - </h1>` berfungsi menampilkan judul utama halaman dengan gaya yang sesuai (kelas page-title dapat diatur melalui CSS).

Artikel pertama sama dengan artikel kedua berfungsi untuk menjelaskan tentang pesan moral dari cerita rakyat setiap artikel memiliki kode *Heading* `<h2>`: Menampilkan subjudul artikel pertama. *Paragraf* `<p>`: Menyediakan deskripsi atau informasi rinci mengenai Tari Pattuddu dan pesan moral yang terkandung dan `<hr class="my-4">`: Garis horizontal yang memberi pemisahan visual antar konten untuk memberi ruang dan struktur yang lebih baik.

Struktur CSS kode ini memanfaatkan *Bootstrap* untuk tata letak responsif, dengan kelas seperti: *container* untuk membuat tata letak halaman responsif. mt-5 untuk memberikan *margin* atas. mb-4 untuk memberikan margin bawah untuk artikel. my-4 untuk margin atas dan bawah pada elemen garis pemisah (`<hr>`).

e. Halaman Tentang dan Hubungi

Pengguna memilih menu tentang. Menampilkan teks informasi serta menghubungi untuk memberikan saran dll. Pengguna dapat membaca latar belakang aplikasi dan menghubungi pengembangan jika perlu.

Gambar 4.69 Halaman Tentang dan Hubungi

```
<div class="container mt-5">
  <h1 class="page-title">Tentang Kami</h1>
  <p>
    Selamat datang di aplikasi Cerita Rakyat! Kami berkomitmen untuk melestarikan dan
    membagikan kekayaan budaya cerita rakyat Nusantara kepada dunia. Melalui aplikasi
    ini, Anda dapat menjelajahi cerita tentang tari pattuddu dan tradisi sayyng
    pattuddu yang kaya akan nilai-nilai luhur dan kebijaksanaan tradisional.
  </p>
  <p>
    Jika Anda memiliki pertanyaan, saran, atau ingin berkontribusi, jangan ragu untuk
    menghubungi kami melalui form di bawah ini.
  </p>
  <?php if (isset($_GET['status']) && $_GET['status'] == 'success'): ?>
    <div class="alert alert-success" role="alert">
      Pesan Anda sudah terkirim. Terima kasih telah menghubungi kami!
    </div>
  <?php endif; ?>

  <h2 id="hubungi">Hubungi Kami</h2>
  <form action="process_contact.php" method="post">
    <div class="form-group">
      <label for="name">Nama</label>
      <input type="text" class="form-control" id="name" name="name" required>
    </div>
    <div class="form-group">
      <label for="email">Email</label>
      <input type="email" class="form-control" id="email" name="email" required>
    </div>
  </form>
</div>
```

```

</div>
<div class="form-group">
  <label for="subject">Subject</label>
  <input type="subject" class="form-control" id="subject" name="subject" required>
</div>
<div class="form-group">
  <label for="message">Pesan</label>
  <textarea class="form-control" id="message" name="message" rows="5"
    required></textarea>
</div>
<button type="submit" class="btn btn-primary">Kirim Pesan</button>
</form>
</div>

```

Gambar 4. 71 *Codingan* Halaman Tentang

```

body {
  font-family: 'Arial', sans-serif; /* Font utama halaman */
  background-color: #f0f2f5; /* Warna latar belakang abu-abu terang */
  color: #333; /* Warna teks abu-abu gelap */
}
.container {
  margin-top: 50px; /* Jarak bagian atas halaman */
  padding: 15px; /* Ruang dalam container */
}
.page-title {
  text-align: center; /* Teks judul di tengah */
  font-size: 2.5rem; /* Ukuran font besar untuk judul */
  color: #343a40; /* Warna abu-abu gelap */
  font-weight: bold; /* Teks judul tebal */
  margin-top: -30px; /* Mengurangi jarak atas */
  margin-bottom: 20px; /* Jarak bawah untuk judul */
}
@media (max-width: 768px) {
  .page-title {
    font-size: 2rem; } /* Ukuran font lebih kecil untuk layar sedang */
}
@media (max-width: 576px) {
  .page-title {
    font-size: 1.8rem; } /* Ukuran font lebih kecil untuk layar kecil */
}
/* Formulir Kontak */
.form-group label {
  font-size: 1rem; /* Ukuran font label form */
  color: #333; /* Warna teks label */
}
.form-control {
  border: 1px solid #ddd; /* Warna border abu-abu lembut */
  border-radius: 5px; /* Membuat sudut melengkung */
  padding: 10px; /* Ruang dalam input */
  font-size: 1rem; /* Ukuran teks input */
}
textarea.form-control {
  resize: none; /* Tidak bisa mengubah ukuran textarea */
}
.btn-primary {
  background-color: #007bff; /* Warna tombol biru */
  border: none; /* Menghapus border */
  font-size: 1rem; /* Ukuran teks tombol */
  padding: 10px 20px; /* Padding dalam tombol */
  border-radius: 5px; /* Membuat sudut tombol melengkung */
  transition: background-color 0.3s ease; /* Efek transisi warna */
}
.btn-primary:hover {
  background-color: #0056b3; /* Warna tombol saat di-hover */
}

```

Gambar 4. 70 *CSS* Halaman Tentang

1. Penjelasan Kode Halaman Pesan Moral

Kode di atas digunakan untuk membuat halaman web yang menampilkan informasi serta menghubungi untuk memberikan saran dll. Berikut adalah penjelasan singkat dan detail mengenai kode tersebut: struktur halaman pada kode `<div class="container mt-5">` yeg terdiri dari `class="container"`: Membuat tata letak yang terpusat dan terstruktur. `mt-5`: Menambahkan *margin* atas yang besar (5 tingkat *Bootstrap*) untuk memberi jarak dari elemen sebelumnya.

Kode Judul Halaman (`<h1 class="page-title">`) `class="page-title"`: Menandai elemen sebagai judul utama dengan gaya khusus misalnya ukuran *font* yang *bold*). Paragraf *Deskripsi* (`<p>`) paragraf ini menjelaskan tujuan aplikasi, yaitu melestarikan budaya cerita rakyat dan mengundang pengguna untuk berkontribusi. Kode pesan sukses PHP (`<?php if ... ?>`) untuk mengecek apakah parameter URL `status=success` tersedia. Jika ya, menampilkan pesan sukses dengan kelas `alert alert-success`, yang menampilkan notifikasi hijau dari *Bootstrap*.

Kode Subjudul Hubungi Kami (`<h2 id="hubungi">`) `id="hubungi"` untuk memberikan penanda unik pada elemen ini, yang memungkinkan tautan langsung menuju bagian "Hubungi Kami". Formulir Kontak (`<form>`) `action="process_contact.php"`: Formulir mengirimkan data ke file `process_contact.php` untuk diproses. `method="post"`: Data dikirim menggunakan metode *POST* untuk keamanan lebih baik.

Kode *Input Formulir* (`<div class="form-group">`) terdiri dari kode `<label>`: Memberikan label untuk setiap input. `<input>`: Menerima data pengguna (nama, *email*, *subjek*). `class="form-control"`: Gaya *Bootstrap* untuk input yang responsif.

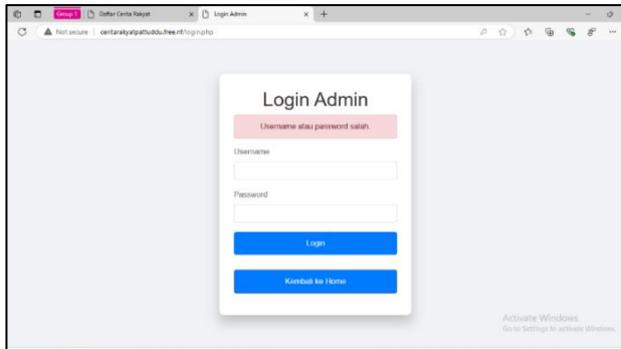
required: Memastikan input ini tidak kosong sebelum pengiriman. `<textarea>`:
Input untuk pesan dengan baris yang dapat diatur. Tombol Kirim (`<button>`)
class="btn btn-primary": Menggunakan gaya tombol biru Bootstrap. Teks tombol
 bertuliskan "Kirim Pesan".

C. Pengujian Sistem

Pengujian sistem pada penelitian ini menggunakan 2 metode yaitu *Black Box Testing* dan *White Box Testing*. Berikut ini hasil pengujian menggunakan sistem tersebut.

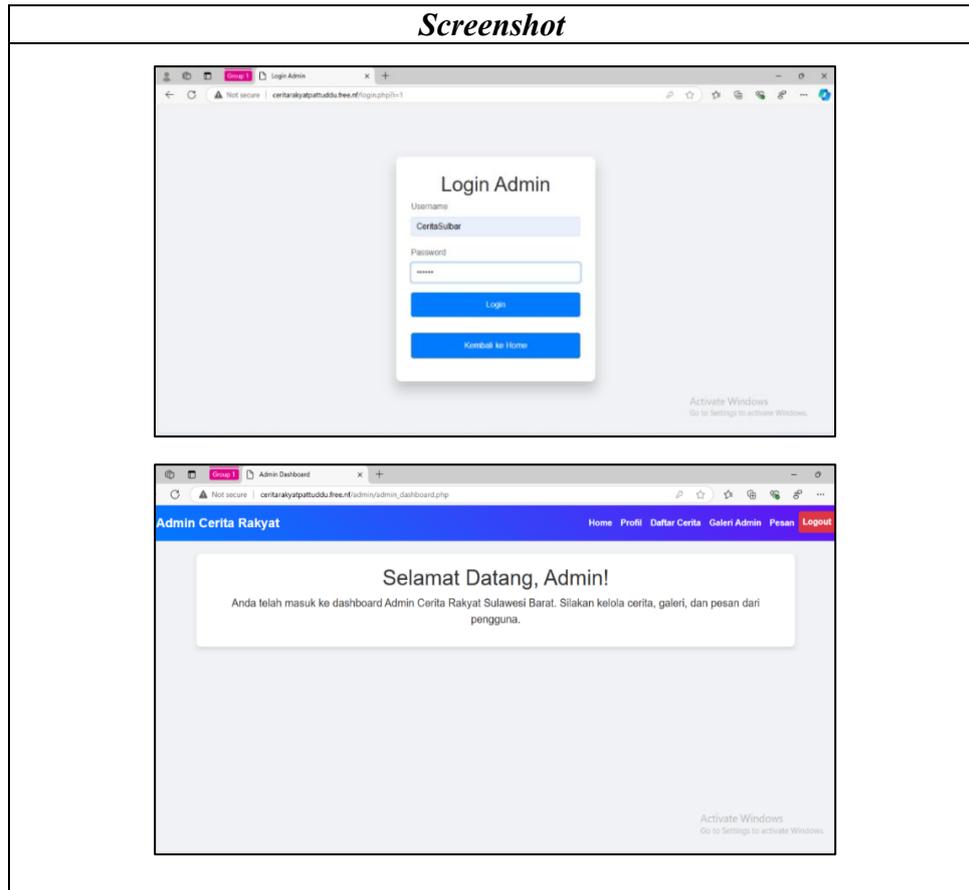
2. Black Box Testing

a. Black Box Testing Kesalahan Username dan Password

Tes Faktor	Hasil	Keterangan
Memasukkan Username atau Password yang tidak sesuai	✓	Berhasil, ketika username atau password tidak sesuai maka tampil pesan Username dan Password Salah!
Screenshot		
		

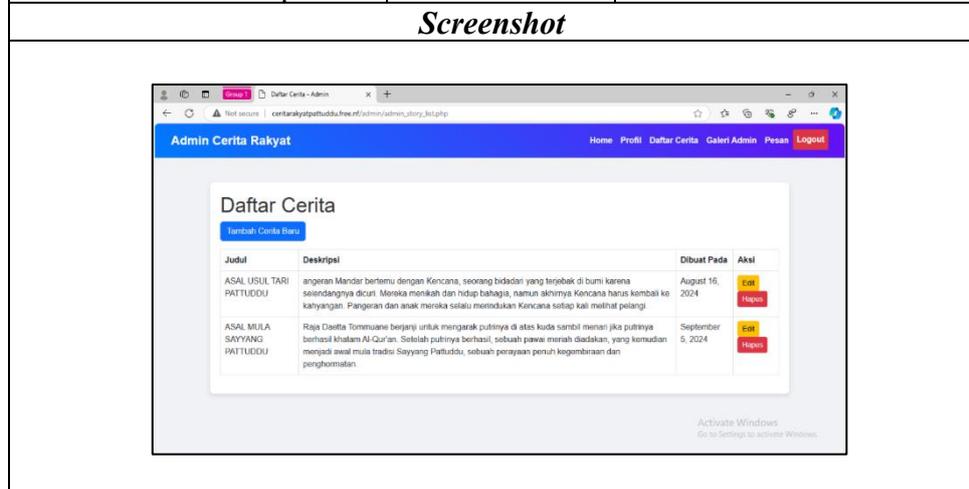
b. Black Box Testing Admin Berhasil Login

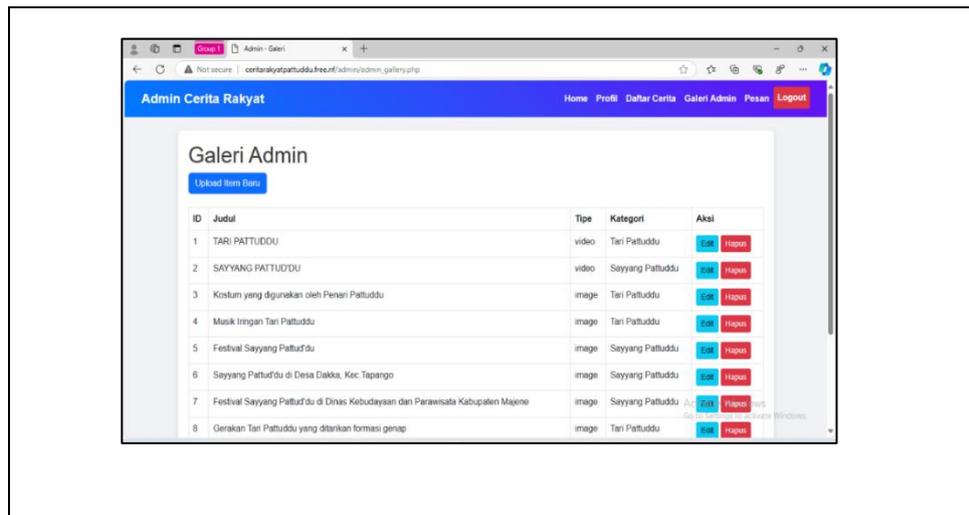
Tes Faktor	Hasil	Keterangan
Memasukkan Username atau Password yang sesuai	✓	Berhasil, sistem menampilkan halaman dashboard Admin



c. *Black Box Testing* Tambah

Tes Faktor	Hasil	Keterangan
Admin Mengisi <i>form</i> upload dan menekan tombol Tambah/Upload	✓	Berhasil, sistem menampilkan data yang berhasil ditambahkan

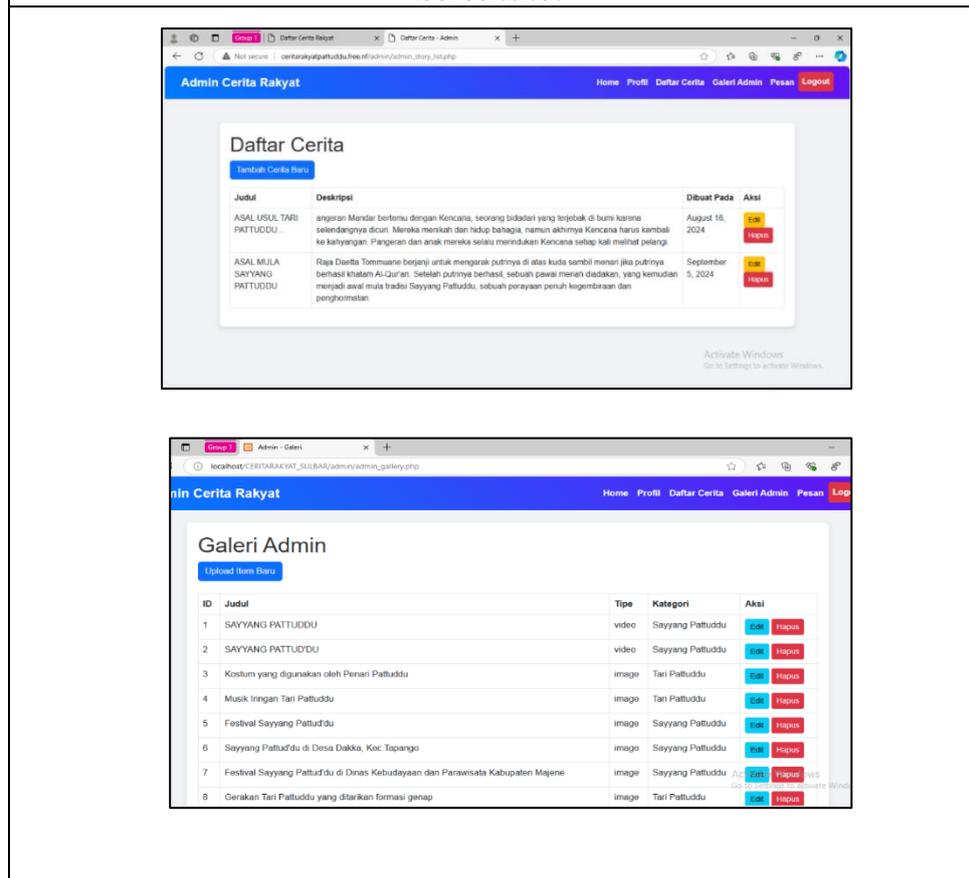




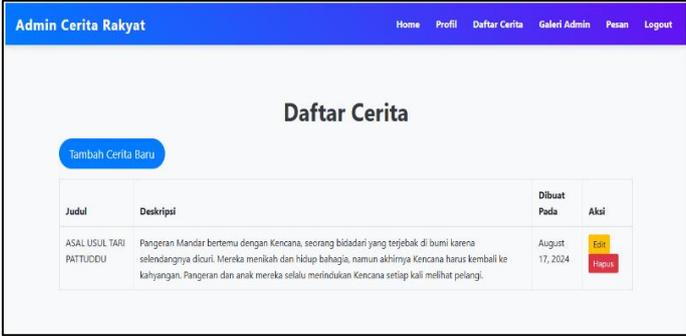
d. Black Box Testing Edit

Tes Faktor	Hasil	Keterangan
Admin Mengisi form edit dan menekan tombol perbarui	✓	Berhasil, sistem menampilkan data yang berhasil diubah

Screenshot

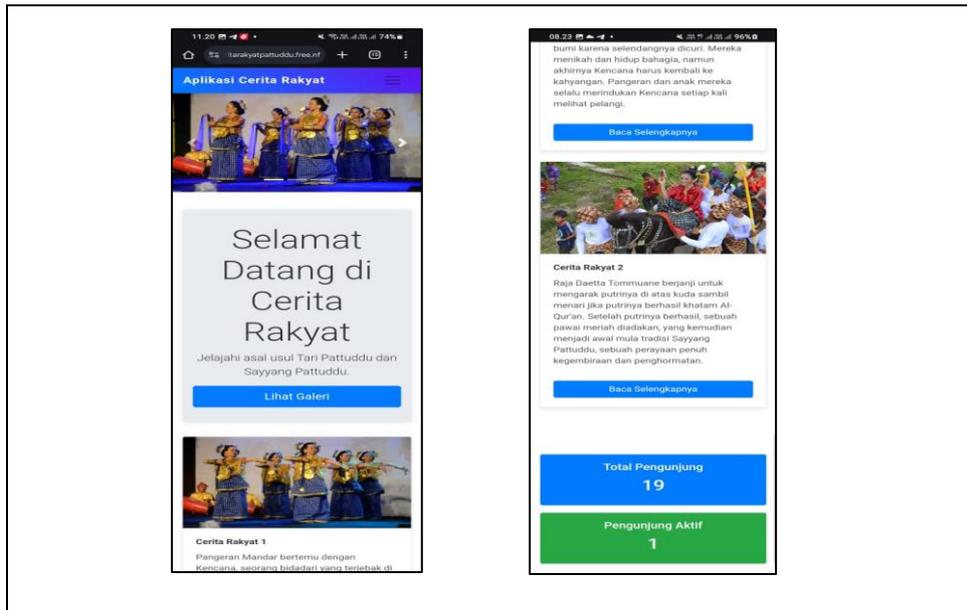


e. *Black Box Testing Hapus*

Tes Faktor	Hasil	Keterangan
Admin menekan tombol Hapus	✓	Berhasil, sistem menampilkan data yang berhasil dihapus
Screenshot		
 		

f. *Black Box Testing Halaman Dashboard User*

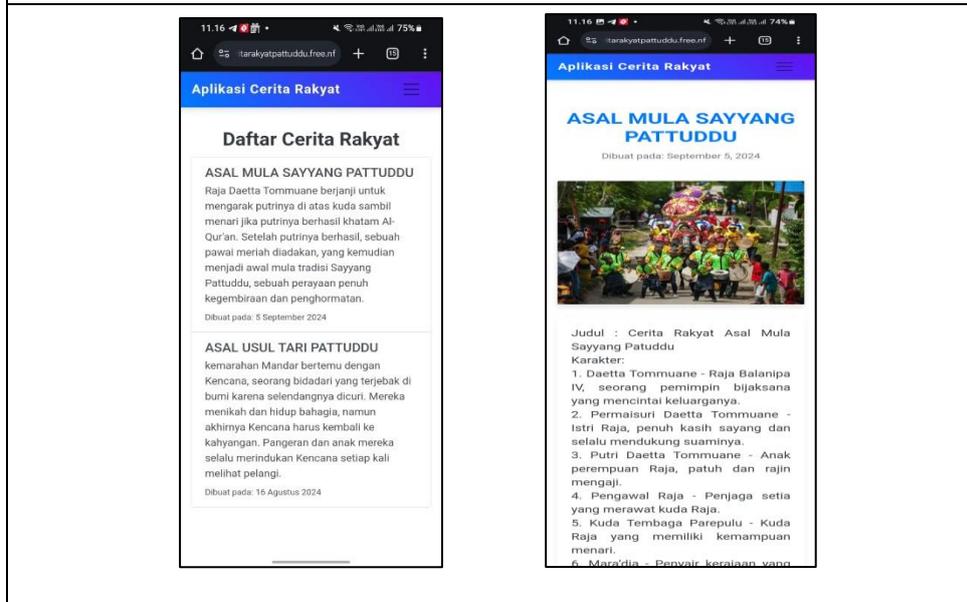
Tes Faktor	Hasil	Keterangan
User membuka halaman dashboard	✓	Berhasil, sistem menampilkan halaman dashboard
Screenshot		



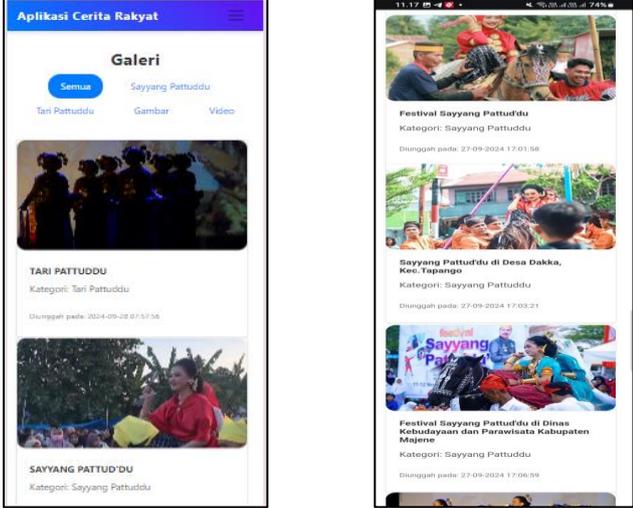
g. *Black Box Testing Halaman Cerita User*

Tes Faktor	Hasil	Keterangan
User memilih menu cerita kemudian pilih cerita yang mau dibaca atau dilihat	✓	Berhasil, sistem menampilkan daftar cerita dan menampilkan cerita yang dipilih

Screenshot



h. *Black Box Testing Halaman Galeri User*

Tes Faktor	Hasil	Keterangan
User memilih menu galeri	✓	Berhasil, sistem menampilkan gambar dan video
Screenshot		
		

i. *Black Box Texting* Halaman Pesan Moral User

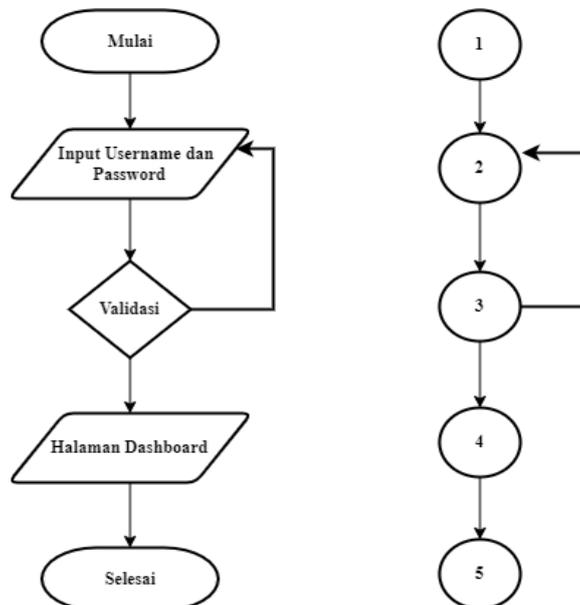
Tes Faktor	Hasil	Keterangan
User memilih menu pesan moral	✓	Berhasil, sistem menampilkan pesan moral dari cerita
Screenshot		
		

j. *Black Box Testing* Halaman Tentang User

Tes Faktor	Hasil	Keterangan
User memilih menu Tentang	✓	Berhasil, sistem menampilkan tentang dan hubungi
Screenshot		
		

2. *White Box Testing*

a. *White Box Testing* Kesalahan Username dan Password



Gambar 4. 72 *Flowchart* Kesalahan Username dan Password

1) Menghitung *cyclomatic complexity* $V(G)$ pada egde dan node

menggunakan rumus: $V(G) = E - N + 2$

$$E (\text{edge}) = 5$$

$$N (\text{node}) = 5$$

$$P (\text{Predikat node}) = 1$$

$$\text{Penyelesaian: } V(G) = E - N + 2$$

$$= 5 - 5 + 2$$

$$= 2$$

$$\text{Predikat (P)} = P + 1$$

$$= 1 + 1$$

$$= 2$$

2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *flowgraph* di

atas memiliki Region = 2

3) Independent path pada *flowgraph* tersebut yakni:

$$\text{Path 1} = 1 - 2 - 3 - 2$$

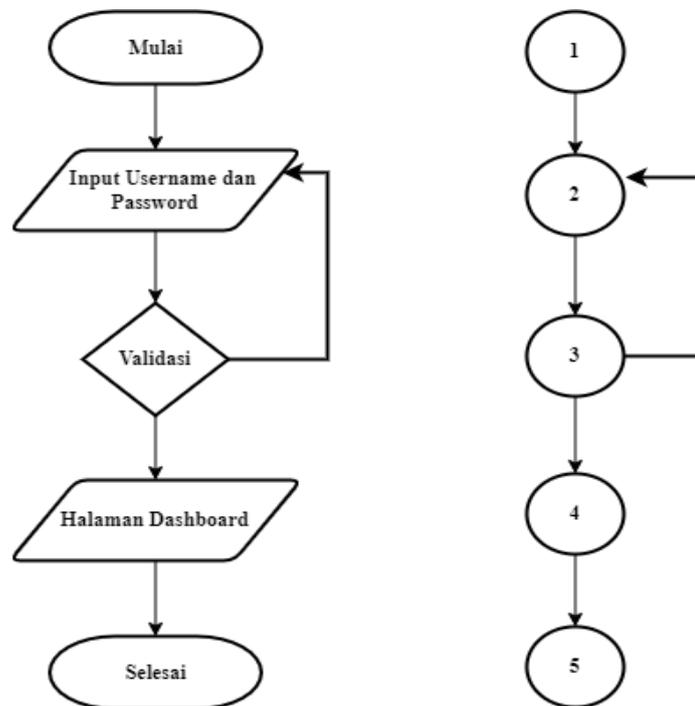
$$\text{Path 2} = 1 - 2 - 3 - 4 - 5$$

4) Grafik matriks

Tabel 4. 5 Grafik Matriks Kesalahan *Username* dan *Password*

	1	2	3	4	5	E-1
1		1				1 - 1 = 0
2			1			1 - 1 = 0
3		1		1		2 - 1 = 1
4					1	1 - 1 = 0
5						0
	SUM (E + 1)					1 + 1 = 2

b. *White Box Testing Login Berhasil Username dan Password*



Gambar 4. 73 *Flowchart dan Flowgraph Login Berhasil*

1) Menghitung *cyclomatic complexity* $V(G)$ pada egde dan node,

menggunakan rumus: $V(G) = E - N + 2$

$$E (\text{edge}) = 5$$

$$N (\text{node}) = 5$$

$$P (\text{Predikat node}) = 1$$

$$\text{Penyelesaian: } V(G) = E - N + 2$$

$$= 5 - 5 + 2$$

$$= 2$$

$$\text{Predikat (P)} = P + 1$$

$$= 1 + 1$$

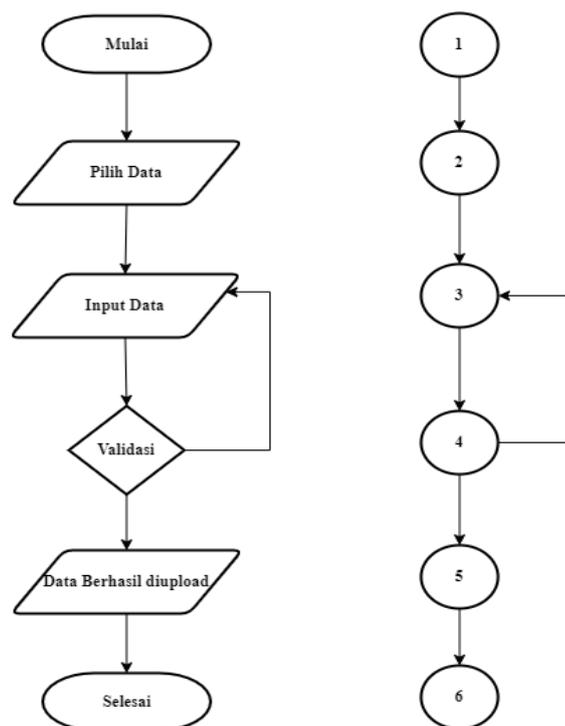
$$= 2$$

- 2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *flowgraph* di atas memiliki Region = 2
- 3) Independent path pada *flowgraph* tersebut yakni:
- Path 1 = 1 – 2 – 3 – 2
- Path 2 = 1 – 2 – 3 – 4 – 5
- 4) Grafik matriks

Tabel 4. 6 Grafik Matriks Berhasil Login

	1	2	3	4	5	E-1
1		1				$1 - 1 = 0$
2			1			$1 - 1 = 0$
3		1		1		$2 - 1 = 1$
4					1	$1 - 1 = 0$
5						0
	SUM (E + 1)					$1 + 1 = 2$

c. *White Box Testing Upload*



Gambar 4. 74 Flowchart dan Flowgraph Upload

1) Menghitung *cyclomatic complexity* $V(G)$ pada egde dan node,

menggunakan rumus: $V(G) = E - N + 2$

$$E (\text{edge}) = 5$$

$$N (\text{node}) = 5$$

$$P (\text{Predikat node}) = 1$$

$$\text{Penyelesaian: } V(G) = E - N + 2$$

$$= 6 - 6 + 2$$

$$= 2$$

$$\text{Predikat (P)} = P + 1$$

$$= 1 + 1$$

$$= 2$$

2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *flowgraph* di

atas memiliki Region = 2

3) Independent path pada flowgraph tersebut yakni:

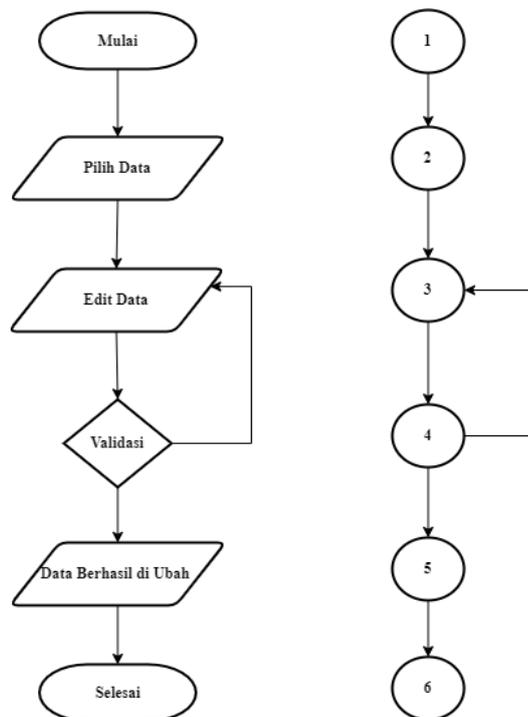
$$\text{Path 1} = 1 - 2 - 3 - 4 - 3$$

$$\text{Path 2} = 1 - 2 - 3 - 4 - 5 - 6$$

4) Grafik matriks

Tabel 4. 7 Grafik Matriks *Upload*

	1	2	3	4	5	6	E-1
1		1					1 - 1 = 0
2			1				1 - 1 = 0
3				1			1 - 1 = 0
4			1		1		2 - 1 = 1
5						1	1 - 1 = 0
6							0
	SUM (E + 1)						1 + 1 = 2

d. *White Box Testing Edit*

Gambar 4. 75 *Flowchart dan Flowgraph Edit*

1) Menghitung *cyclomatic complexity* $V(G)$ pada egde dan node,

menggunakan rumus: $V(G) = E - N + 2$

$$E \text{ (edge)} = 5$$

$$N \text{ (node)} = 5$$

$$P \text{ (Predikat node)} = 1$$

$$\text{Penyelesaian: } V(G) = E - N + 2$$

$$= 6 - 6 + 2$$

$$= 2$$

$$\text{Predikat (P)} = P + 1$$

$$= 1 + 1$$

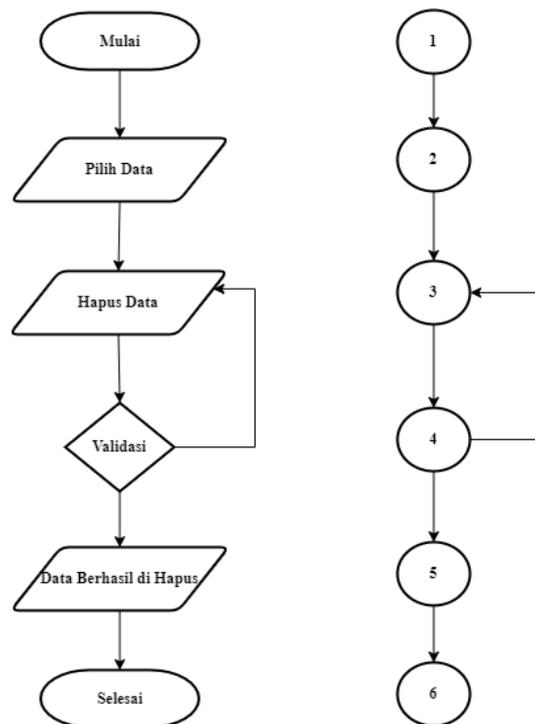
$$= 2$$

- 2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *flowgraph* di atas memiliki Region = 2
- 3) Independent path pada *flowgraph* tersebut yakni:
 Path 1 = 1 – 2 – 3 – 4 – 3
 Path 2 = 1 – 2 – 3 – 4 – 5 – 6
- 4) Grafik matriks

Tabel 4. 8 Grafik Matriks Edit

	1	2	3	4	5	6	E-1	
1		1					1 - 1 = 0	
2			1				1 - 1 = 0	
3				1			1 - 1 = 0	
4			1		1		2 - 1 = 1	
5						1	1 - 1 = 0	
6							0	
	SUM (E + 1)							1 + 1 = 2

e. *White Box Testing* Hapus



Gambar 4. 76 *Flowchart* dan *Flowgraph* Hapus

1) Menghitung *cyclomatic complexity* $V(G)$ pada egde dan node,

menggunakan rumus: $V(G) = E - N + 2$

$$E (\text{edge}) = 5$$

$$N (\text{node}) = 5$$

$$P (\text{Predikat node}) = 1$$

$$\text{Penyelesaian: } V(G) = E - N + 2$$

$$= 6 - 6 + 2$$

$$= 2$$

$$\text{Predikat (P)} = P + 1$$

$$= 1 + 1$$

$$= 2$$

2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *flowgraph* di

atas memiliki Region = 2

3) Independent path pada *flowgraph* tersebut yakni:

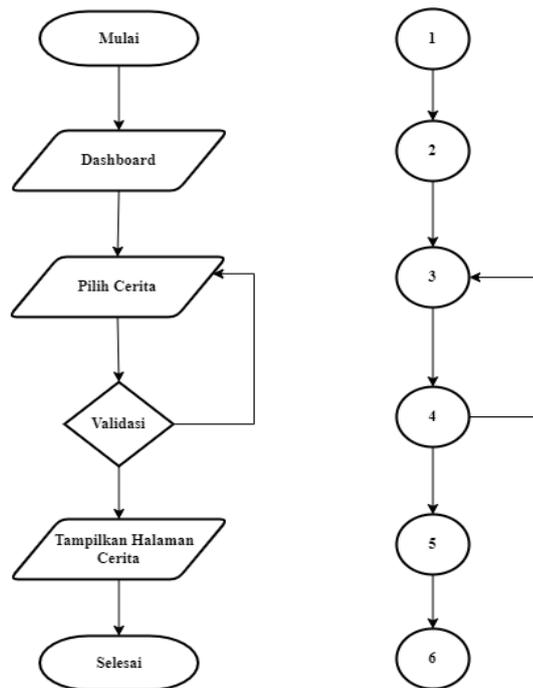
$$\text{Path 1} = 1 - 2 - 3 - 4 - 3$$

$$\text{Path 2} = 1 - 2 - 3 - 4 - 5 - 6$$

4) Grafik matriks

Tabel 4. 9 Grafik Matriks Hapus

	1	2	3	4	5	6	E-1	
1		1					$1 - 1 = 0$	
2			1				$1 - 1 = 0$	
3				1			$1 - 1 = 0$	
4			1		1		$2 - 1 = 1$	
5						1	$1 - 1 = 0$	
6							0	
	SUM (E + 1)							$1 + 1 = 2$

f. *White Box Testing* Halaman Cerita

Gambar 4. 77 *Flowchart* dan *Flowgraph* Halaman Cerita

1) Menghitung *cyclomatic complexity* $V(G)$ pada edge dan node,

menggunakan rumus: $V(G) = E - N + 2$

$$E \text{ (edge)} = 5$$

$$N \text{ (node)} = 5$$

$$P \text{ (Predikat node)} = 1$$

$$\text{Penyelesaian: } V(G) = E - N + 2$$

$$= 6 - 6 + 2$$

$$= 2$$

$$\text{Predikat (P)} = P + 1$$

$$= 1 + 1$$

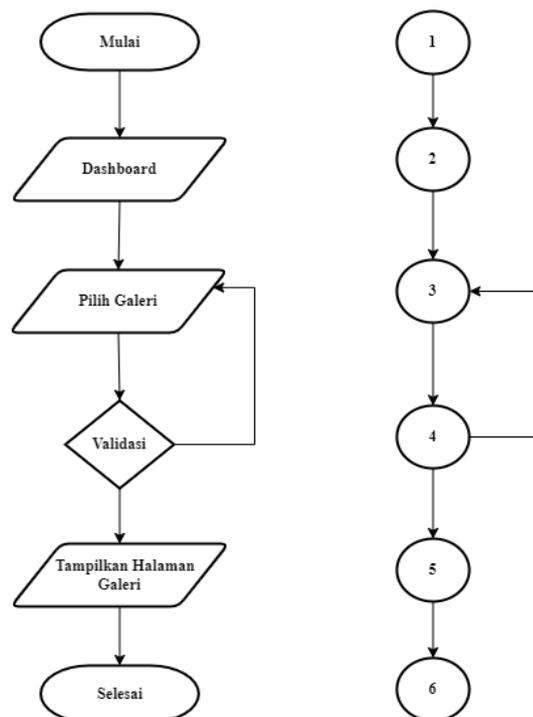
$$= 2$$

- 2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *flowgraph* di atas memiliki Region = 2
- 3) Independent path pada *flowgraph* tersebut yakni:
- Path 1 = 1 – 2 – 3 – 4 – 3
- Path 2 = 1 – 2 – 3 – 4 – 5 – 6
- 4) Grafik matriks

Tabel 4. 10 Grafik Matriks Halaman Cerita

	1	2	3	4	5	6	E-1	
1		1					$1 - 1 = 0$	
2			1				$1 - 1 = 0$	
3				1			$1 - 1 = 0$	
4			1		1		$2 - 1 = 1$	
5						1	$1 - 1 = 0$	
6							0	
	SUM (E + 1)							$1 + 1 = 2$

- g. *White Box Testing* Halaman Galeri



Gambar 4. 78 *Flowchart* dan *Flowgraph* Halaman Galeri

5) Menghitung *cyclomatic complexity* $V(G)$ pada egde dan node,

menggunakan rumus: $V(G) = E - N + 2$

$$E (\text{edge}) = 5$$

$$N (\text{node}) = 5$$

$$P (\text{Predikat node}) = 1$$

$$\text{Penyelesaian: } V(G) = E - N + 2$$

$$= 6 - 6 + 2$$

$$= 2$$

$$\text{Predikat (P)} = P + 1$$

$$= 1 + 1$$

$$= 2$$

6) Berdasarkan perhitungan *Cyclomatic Complexity* dari *flowgraph* di

atas memiliki Region = 2

7) Independent path pada *flowgraph* tersebut yakni:

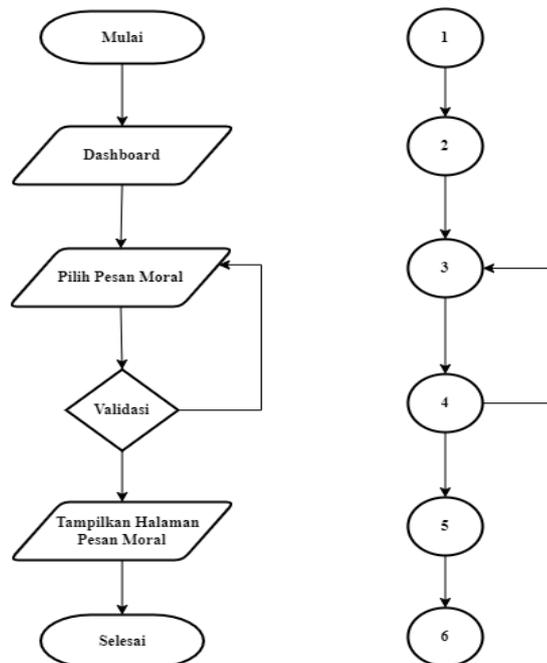
$$\text{Path 1} = 1 - 2 - 3 - 4 - 3$$

$$\text{Path 2} = 1 - 2 - 3 - 4 - 5 - 6$$

8) Grafik matriks

Tabel 4. 11 Grafik Matriks Halaman Galeri

	1	2	3	4	5	6	E-1	
1		1					$1 - 1 = 0$	
2			1				$1 - 1 = 0$	
3				1			$1 - 1 = 0$	
4			1		1		$2 - 1 = 1$	
5						1	$1 - 1 = 0$	
6							0	
	SUM (E + 1)							$1 + 1 = 2$

h. *White Box Testing* Halaman Pesan Moral

Gambar 4. 79 Flowchart dan Flowgraph Halaman Pesan Moral

1) Menghitung *cyclomatic complexity* $V(G)$ pada egde dan node,

menggunakan rumus: $V(G) = E - N + 2$

$$E \text{ (edge)} = 5$$

$$N \text{ (node)} = 5$$

$$P \text{ (Predikat node)} = 1$$

$$\text{Penyelesaian: } V(G) = E - N + 2$$

$$= 6 - 6 + 2$$

$$= 2$$

$$\text{Predikat (P)} = P + 1$$

$$= 1 + 1$$

$$= 2$$

2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *flowgraph* di atas memiliki Region = 2

3) Independent path pada *flowgraph* tersebut yakni:

Path 1 = 1 – 2 – 3 – 4 – 3

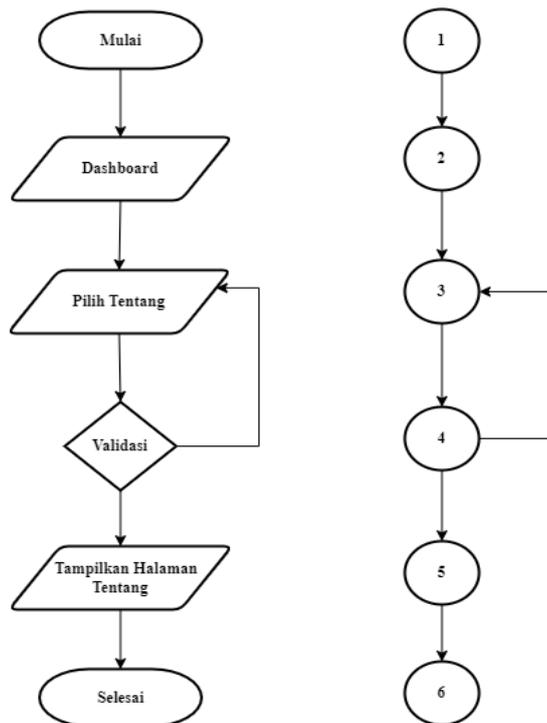
Path 2 = 1 – 2 – 3 – 4 – 5 – 6

4) Grafik matriks

Tabel 4. 12 Grafik Matriks Halaman Pesan Moral

	1	2	3			4	5	6	E-1
1			1					1 - 1 = 0	
2				1				1 - 1 = 0	
3					1			1 - 1 = 0	
4			1			1		2 - 1 = 1	
5							1	1 - 1 = 0	
6								0	
	SUM (E + 1)								1 + 1 = 2

i. *White Box Testing* Halaman Tentang



Gambar 4. 80 Flowchart dan Flowgraph Halaman Tentang

1) Menghitung *cyclomatic complexity* $V(G)$ pada egde dan node,

menggunakan rumus: $V(G) = E - N + 2$

$$E (\text{edge}) = 5$$

$$N (\text{node}) = 5$$

$$P (\text{Predikat node}) = 1$$

$$\text{Penyelesaian: } V(G) = E - N + 2$$

$$= 6 - 6 + 2$$

$$= 2$$

$$\text{Predikat (P)} = P + 1$$

$$= 1 + 1$$

$$= 2$$

2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *flowgraph* di

atas memiliki Region = 2

3) Independent path pada *flowgraph* tersebut yakni:

$$\text{Path 1} = 1 - 2 - 3 - 4 - 3$$

$$\text{Path 2} = 1 - 2 - 3 - 4 - 5 - 6$$

4) Grafik matriks

Tabel 4. 13 Grafik Matriks Halaman Tentang

	1	2	3	4	5	6	E-1
1		1					1 - 1 = 0
2			1				1 - 1 = 0
3				1			1 - 1 = 0
4			1		1		2 - 1 = 1
5						1	1 - 1 = 0
6							0
	SUM (E + 1)						1 + 1 = 2

BAB V

PENUTUP

A. Kesimpulan

Berdasarkan penelitian yang telah dilakukan maka dapat ditarik kesimpulan bahwa perancangan aplikasi cerita rakyat asal usul tari pattuddu, adalah sebagai berikut:

1. Hasil pengujian menunjukkan bahwa aplikasi berfungsi sesuai dengan spesifikasi yang ditetapkan, dengan fitur-fitur seperti teks, animasi cerita rakyat, galeri menampilkan foto serta video terkait budaya dari cerita rakyat, dan pesan moral. Aplikasi ini dapat meningkatkan aksesibilitas dan minat terhadap cerita rakyat, khususnya di era digital.
2. Pengujian sistem *Black Box Testing* dan *White Box Testing* menunjukkan bahwa aplikasi dapat berfungsi sebagaimana mestinya dan beroperasi dengan lancar. dengan menguji sistem, sehingga penggunaan dijamin dapat diandalkan.

B. Saran

Pada penelitian ini penulis menyadari bahwa masih ada beberapa kekurangan yang sangat perlu diperbaiki dan pengembangan di penelitian selanjutnya. Oleh karena itu penulis memiliki saran untuk pengembangan selanjutnya:

1. Aplikasi ini dapat dikembangkan dengan desain interface yang lebih mudah diakses dengan berbagai perangkat, sehingga meningkatkan pengalaman pengguna.
2. Aplikasi ini dapat dikembangkan dengan penambahan media interaktif, seperti kuis dan permainan, yang dirancang untuk menguji pemahaman pengguna. Penambahan ini bertujuan untuk meningkatkan minat dan memperdalam wawasan pengguna tentang cerita rakyat asal-usul tari pattuddu.

DAFTAR PUSTAKA

- Agung, M. (2018). Garudaku: inovasi pelestarian budaya nusantara berbasis cultural maps menuju Indonesia emas 2045. *Jurnal PENA: Penelitian dan Penalaran*, 5(2), 977-988.
- Alimuddin, R. (2011). Mandar: Alam, Budaya, Manusia. *Dinas Perhubungan, Komunikasi dan Informatika Kabupaten Polewali Mandar*. Polewali Mandar : Dinas Perhubungan, Komunikasi dan Informatika.
- Ayu. (2021). Pengertian Flowchart yang Perlu Diketahui, Bedakan Jenis dan Penggunaannya. Diperoleh dari: <https://www.merdeka.com/jateng/pengertian-flowchart-yang-perlu-diketahui-bedakan-jenis-dan-penggunaanya-klm.html>. (Diakses 27 April 2024).
- Ayu, D. (2015). *Cerita Rakyat Nusantara 34 Provinsi*. Jakarta Selatan: Wahyumedia.
- Bakhri, Syamsul. "Perancangan Animasi Interaktif Cerita Rakyat Asal Usul Nama Kota Jambi Berbasis Android." *SATIN-Sains dan Teknologi Informasi* 8.1 (2022): 32-41.
- Destriana, dkk. (2021). Diagram UML Dalam Membuat Aplikasi Android Firebase" Studi Kasus Aplikasi Bank Sampah". Yogyakarta: Deepublish.
- Febyanti, F., Ayuswantana, A. C., & Wulandari, S. (2023). Perancangan Web Komik Angling Dharma Sebagai Pengenalan Cerita Rakyat Jawa Timur Kepada Dewasa 18-25 Tahun.
- Huda, M. (2020). Bootstrap 4: Belajar CRUD Menggunakan PHP dan MySQL.
- Kurniawan, D. (2023). *Belajar Pemrograman Web Dasar HTML, CSS, & Javascript Untuk Pemula*. Penerbit Yayasan Prima Agus Teknik, 1-228.
- Limbong, T. (2021). *Pemrograman Web Dasar*. Yayasan Kita Menulis
- Ma`lum Rasyid. (2016). *Saiyyang Pattudung dan Khataman Al-Qur'an di Mandar*. Solo: Zadhaniva Publishing.
- Microsoft. (2024). *Visual Studio Code Documentation*. Diakses dari <https://code.visualstudio.com/docs>. (Diakses 29 juni 2024).
- Ramadhani, Halida Dwi Edyati, Nugrahardi. 2020. "Perancangan Animasi 2D "Pangeran Lembu." *Jurnal Sains Dan Seni ITS*. Vol. 9, No. 2 (2020), 2337-3520 (2301-928X) 9 (2).

- Reggy, C. (2022). *Ilustrasi Desain Dalam Media Interaktif Cerita Rakyat Sukabumi Berbasis Aplikasi Untuk Anak Sekolah Dasar Di Sd Islam Attarbiyah* (Doctoral dissertation, *Nusa Putra*).
- Roosendal. T. (2022). *About Blender*. Diperoleh Oleh: <https://www.blender.org/about/>. (Diakses 6 Mei 2024).
- Sari, A. O., & Abdilah, A. Sunarti. (2019). *Web Programming*. Graha Ilmu.
- Wikipedia. (2024). *Blender 3D: Noob to Pro*. https://en.wikibooks.org/wiki/Blender_3D:_Noob_to_Pro (Diakses 08 September 2024).
- Zulkiram, Ahmad, Ahmad M. Sewang, and G. Wahyuddin. "Tradisi Messawe To Tamma'di Desa Pambusuang Kecamatan Balanipa Kabupaten Polewali Mandar (Studi Budaya Islam)." *El-Fata: Journal of Sharia Economics and Islamic Education* 2.2 (2023): 120-139.