

PENERAPAN REST API SISTEM INFORMASI CINEMA CATALOG MOVIE BERBASIS DESKTOP DAN WEB SERVICE

IMPLEMENTATION OF REST API FOR DESKTOP AND WEB SERVICE-BASED CINEMA CATALOG MOVIE INFORMATION SYSTEM

Mohammad Farizd^{1*}, Aditya Kurnia Pratama¹, Abdul Rezha Efrat Najaf¹

*E-mail: mohammadfarizd4@gmail.com

¹Sistem Informasi, Fakultas Ilmu Komputer, UPN “Veteran” Jawa Timur

Abstrak

Banyaknya pembuatan sistem yang berbasis Desktop dan Web Client membuat pengguna kebingungan website mana yang harus dikunjungi dalam mendapatkan informasi dari suatu film yang sedang dicari. Lebih daripada itu, informasi yang diberikan masih minim, sehingga membuat tidak efektif dalam memaksimalkan penggunaan waktu. Dengan adanya perkembangan teknologi saat ini, aplikasi berhasil dibuat yang berbasiskan Desktop dan Website. Metode pengumpulan data ini digunakan sebagai penunjang dan arahan dalam penelitian ini. Selain itu, digunakan juga metode pengembangan sistem menggunakan model metode System Development Life Cycle (SDLC), khususnya model metode waterfall, yang merupakan pengembangan sistem informasi yang berurutan, sistematis, dan berulang. Penelitian ini menghasilkan rancangan sistem informasi Cinema Catalog Movie dengan menggunakan beberapa diagram untuk menggambarkan dan mengorganisir fungsionalitas sistem, antara lain Use Case Diagram, Class Diagram, Activity Diagram, dan Sequence Diagram. Implementasi sistem ini menghasilkan tampilan website dan desktop dengan hak akses admin, yang memungkinkan pengelolaan dan penyajian informasi mengenai katalog film.

Kata kunci: *REST API, Sistem Informasi, Desktop, Web Service*

Abstract

The large number of desktop and web client-based systems makes users confused about which website to visit to get information about the film they are looking for. More than that, the information provided is still minimal, thus making it ineffective in maximizing time. With current technological developments, applications have been successfully created that are desktop and website based. This data collection method is used as support and direction in this study. In addition, system development methods are also utilized using the System Development Life Cycle (SDLC) model, especially the waterfall method model, which is a sequential, systematic, and iterative information system development. This research resulted in a Cinema Catalog Movie information system design using several diagrams to describe and organize system functionality, including Use Case Diagrams, Class Diagrams, Activity Diagrams, and Sequence Diagrams. The implementation of this system produces a website and desktop display with admin access rights, which allows the management and presentation of information about film catalogs.

Keywords: *REST API, Information System, Desktop, Web Service*

1. PENDAHULUAN

Di era digital Perkembangan teknologi yang semakin maju, integrasi sistem informasi menjadi kunci utama bagi organisasi untuk mencapai efisiensi dan produktivitas yang tinggi [1]. Sistem informasi yang Putus-putus dapat menghambat alur kerja dan menghambat akses data yang diperlukan untuk pengambilan keputusan yang cepat dan akurat. hampir seluruh kegiatan dan

aktivitas kehidupan manusia tidak terlepas dari namanya Internet, baik di bidang pendidikan, hiburan, dan lain sebagainya.

Dalam konteks industri hiburan, khususnya di industri film, manajemen dan akses terhadap data film menjadi hal yang krusial. Sebuah perusahaan produksi film mungkin memiliki beberapa sistem informasi terpisah seperti sistem manajemen inventaris film, sistem pemesanan tiket online, dan sistem manajemen penjualan merchandise [2]. Maka dari itu, diperlukan suatu solusi yang dapat mengintegrasikan sistem-sistem tersebut agar dapat saling berkomunikasi dan berbagi data dengan mudah. Film merupakan sebuah media komunikasi berbentuk audio visual yang tidak hanya memberikan hiburan, tetapi juga dapat menjadi sarana pembelajaran melalui cerita yang disampaikannya. Minat terhadap media ini tidak terbatas pada industri film [3] Film katalog sendiri memiliki nilai signifikan dalam industri hiburan. Tidak hanya memberikan hiburan kepada penonton, tetapi juga memberikan peluang bagi studio produksi dan distributor untuk memperoleh pendapatan tambahan. Film-film dalam katalog dapat didistribusikan melalui berbagai saluran, seperti teater bioskop, penjualan DVD atau Blu-ray, penyiaran televisi, platform streaming online, dan lain sebagainya.

Integrasi sistem mengarah pada proses menghubungkan dan mengkoordinasikan sistem-sistem yang berbeda agar dapat beroperasi secara terpadu. Dalam konteks ini, integrasi sistem informasi mengacu pada menghubungkan dan mengintegrasikan berbagai sistem informasi yang ada dalam suatu organisasi atau lingkungan kerja. Hal ini bertujuan untuk meningkatkan aliran informasi, efisiensi operasional, dan memungkinkan berbagi data yang konsisten antara sistem-sistem tersebut. Pemrograman desktop mengarah pada pengembangan perangkat lunak yang dijalankan pada komputer atau perangkat lokal. Biasanya, aplikasi desktop memiliki antarmuka pengguna yang lebih lengkap dan kaya fitur, karena memiliki akses penuh ke sumber daya komputer lokal. Pemrograman desktop menggunakan bahasa pemrograman dan kerangka kerja tertentu untuk membangun aplikasi yang dapat diinstal dan dijalankan pada komputer pengguna [4].

Web service adalah metode komunikasi antara dua atau lebih sistem yang berbeda melalui internet. Layanan web memungkinkan sistem ini untuk berkomunikasi dan bertukar data menggunakan protokol standar seperti HTTP [10]. Web service sering digunakan dalam integrasi sistem, di mana satu sistem dapat memanggil layanan web dari sistem lain untuk mengakses dan berbagi data. REST API yaitu gaya arsitektur perangkat lunak yang memungkinkan komunikasi antara sistem-sistem yang terdistribusi melalui protokol HTTP [5]. Dengan menggunakan REST API, sistem-sistem yang berbeda dapat berinteraksi dengan data dan layanan yang tersedia secara terstandarisasi dan mudah dipahami [6].

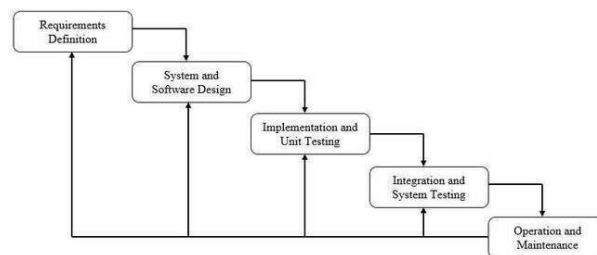
Dengan adanya sistem informasi mengenai katalog film, user dapat mencari dan mengetahui bagaimana rating dan synopsis dari film yang dicari. Maka dari itu dibutuhkan suatu sistem yang saling terintegrasi satu sama lain yang dapat mengelola produksi film secara real time dan akurat, sehingga pengguna film catalog dapat menonton dengan lebih efektif dan efisien. Dalam penelitian ini sistem inventory yang dirancang adalah website dan aplikasi desktop. Selain itu, menurut penulis metode yang paling tepat guna mengintegrasikan sistem adalah dengan metode web service REST, karena metode ini dapat dengan mudah mengintegrasikan antar-sistem dan antar-database. Web service REST juga memiliki performa yang lebih bagus, ringan, mudah, dan hemat bandwidth bila dibandingkan dengan metode lainnya. Data yang diterima akan diubah ke dalam format Retrofit karena Retrofit mudah dibaca dan ditulis [11]. Sistem integrasi data dari Sistem Informasi ini dirancang untuk mengelola dan menyajikan informasi mengenai katalog film untuk memudahkan admin, user untuk melihat data film dan presensi user, serta memudahkan admin dalam melakukan integrasi data jadwal agar Elearning dapat berjalan.

2. METODOLOGI

Pada penelitian ini menggunakan berbagai macam metode untuk pengumpulan data, dimana bertujuan untuk sebagai penunjang dan arahan pada penulisan penelitian ini. Metode

pengumpulan data yang digunakan yakni Studi Pustaka dimana sangat berguna sebagai metode yang tepat terhadap menyelesaikan sebuah permasalahan pada penelitian ini, selain itu digunakan sebagai acuan untuk mendapatkan sebuah informasi yang berkaitan dengan judul penelitian ini. Metode pengumpulan data yang kedua yaitu Observasi, dimana dalam metode ini digunakan untuk mengamati secara keseluruhan mengenai intergrasi sistem informasi yang berhubungan dengan permasalahan sekaligus topik judul dari penelitian ini.

Selain pengumpulan data terdapat juga pengembangan sekaligus perancangan sebuah sistem terhadap penelitian ini menggunakan sebuah salah satu model metode System Development Life Cycle (SDLC), yakni untuk pendukung dalam melakukan pengembangan sebuah perangkat lunak yang akan di rancang dan dibangun. Model metode SDLC ini sering juga dikenal dengan model metode waterfall, dimana metode waterfall merupakan sebuah pengembangan sistem informasi yang prosesnya berurutan, sistematis, dan berulang [7]. Selain itu, model metode Waterfall merupakan sebuah model metode yang sering digunakan sebagai proses pengembangan dan perubahan sistem.



Gambar 1. Alur Model Metode Waterfall

Konsep model metode Waterfall (SDLC) dari pengembangan software terdiri dari berbagai tahapan seperti Analisa Kebutuhan Software, Desain Sistem, Implementasi, Pengujian, dan yang terakhir Pemeliharaan [8].

2.1 Analisa Kebutuhan Perangkat Lunak

Pada tahap ini melakukan Analisa kebutuhan yakni seperti menganalisis data dan informasi, serta mengidentifikasi sebuah permasalahan masalah.

2.1.1 Analisa Data dan Informasi

Sebelum melakukan perancangan sistem informasi yang baru diperlukan menganalisis beberapa data serta informasi yang dibutuhkan untuk menunjang sebuah sistem yang akan dibuat. Pada studi kasus kali ini yakni membuat katalog data dari sebuah identitas sekaligus informasi dari film tersebut. Hal ini digunakan untuk mengetahui bagaimana rancangan yang akan dibuat sekaligus data yang akan ditampilkan.

2.1.2 Analisa Kebutuhan Sistem

Analisa ini digunakan untuk memperkirakan kebutuhan yang digunakan terhadap sistem yang baru. Sistem tersebut meliputi,

2.1.2.1 Perangkat Keras

- Komputer atau Laptop dengan RAM minimal 4GB
- Komputer atau Laptop dengan Harddisk/Solid State Disk minimal 256GB
- Komputer atau Laptop dengan CPU minimal Core 2

2.1.2.2 Perangkat Lunak

- Sistem Operasi Windows sebagai server API Web Service dan server aplikasi Web Client
- Microsoft Visual Basic sebagai aplikasi Desktop
- Web Browser untuk mengakses web client
- Xampp sebagai Web Server
- MySQL sebagai DBMS
- ODBC sebagai penghubung antara Database dengan Microsoft Visual Basic

2.1.2.3 Perangkat Lunak

- Admin, yakni pengguna yang memiliki hak akses control secara penuh terhadap web client dan aplikasi desktop. Kemampuan yang harus dimiliki oleh seorang admin yaitu dapat melakukan pengoperasian secara normal terhadap sistem operasi Windows secara baik, sekaligus dapat mengakses web client yang digunakan.
- User, yakni pengguna yang memiliki hak akses control yang minim dari pada admin. Dimana user ini harus memiliki kemampuan pengoperasian secara baik mengenai akses web client yang akan digunakan.

2.2 Desain Sistem

Pada tahapan ini desain sistem merupakan representasi dari kebutuhan perangkat lunak pada tahap sebelumnya, yang mana nantinya dapat diimplementasikan sebagai perangkat lunak ditahap selanjutnya. Hal ini desain sistem dibuat sesuai dengan UML, meliputi Use Case Diagram, Class Diagram, Activity Diagram, dan Sequence Diagram. Penggunaan Unified Modeling Language (UML) dimaksud yaitu penjelasan yang lebih terperinci dalam perancangan pembuatan program dan rancangan database dari sistem informasi yang akan dibuat [3].

2.3 Implementasi

Tahapan implementasi ini, penulis membangun sistem baru dari hasil desain sistem yang telah di definisikan pada tahap sebelumnya. Hal ini penulis akan membangun sebuah sistem yang berbasis Desktop dan Web Client, dengan mengintegrasikan kedua hal tersebut. Integrasi yang digunakan penulis yaitu menggunakan REST API dengan parameter yang memberikan respon dalam format JSON [9].

2.4 Pengujian

Pada tahap pengujian ini melakukan pengujian terhadap sistem yang telah dibangun agar dapat berjalan secara normal dan fungsional. Dipastikan secara keseluruhan sistem diuji untuk meminimalkan kesalahan (error) pada saat sistem berjalan, sekaligus output yang diharapkan oleh pengguna.

2.5 Pemeliharaan

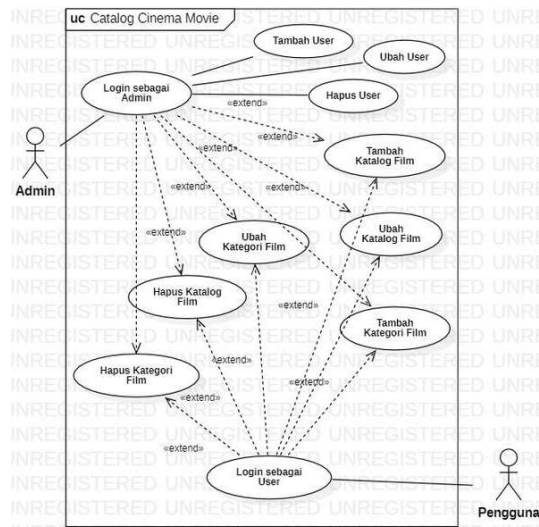
Pada tahap terakhir ini, rancangan sistem informasi Catalog Cinema Movie yang telah digunakan akan selalu diujikan sebelum diberikan kepada pengguna dan dilakukan pemeliharaan secara berkala.

3. HASIL DAN PEMBAHASAN

Penelitian ini menghasilkan beberapa poin yang akan dijelaskan melalui tahapan implementasi dari rancangan sistem informasi Cinema Catalog Movie yang telah dibuat, sebagai berikut.

3.1 Use Case Diagram

Use Case Diagram digunakan untuk menggambarkan secara garis besar dari fungsionalitas aplikasi [9]. Pada use case diagram terdiri dari beberapa actor dan atribut di dalamnya.

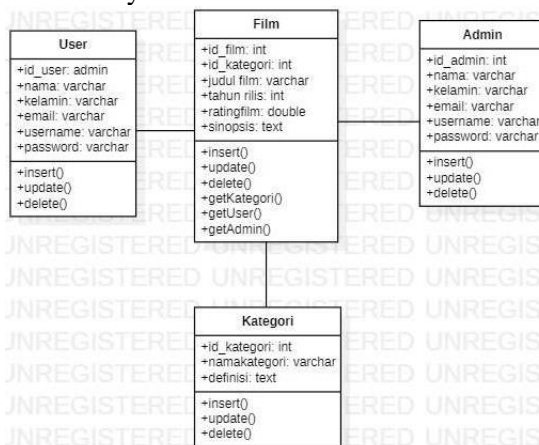


Gambar 2. Use Case Diagram Cinema Catalog Movie

Pada **Gambar 2**, menggambarkan beberapa kegiatan atau hak akses dari masing-masing actor. Dimana pada use case tersebut terdapat dua actor yakni Admin dan Pengguna, dengan hak akses yang berbeda. Perbedaan admin dan pengguna hanya terdapat pada hak akses untuk menambahkan, menghapus, dan mengubah dari data User untuk admin. Sedangkan Pengguna tidak bisa melakukan kegiatan aktivitas tersebut. Namun persamaan hak akses oleh kedua actor yaitu dapat menambahkan, mengubah, dan menghapus data dari film sekaligus data dari katalog film

3.2 Class Diagram

Class Diagram merupakan sebuah kelas-kelas yang akan diimplementasikan menjadi objek atau property terhadap sistem di dalamnya.

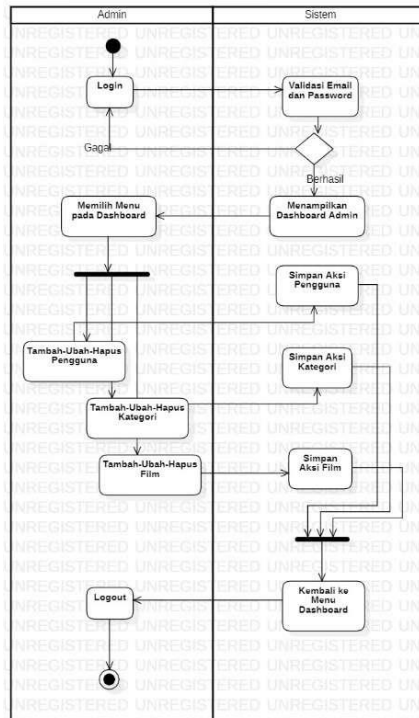


Gambar 3. Class Diagram Cinema Catalog Movie

Pada **Gambar 3**, merupakan sebuah class yang akan di inisiasikan terhadap sistem. Class diagram tersebut terdiri dari Class User, Class Admin, Class Film, dan Class Kategori. Dimana di berbagai class tersebut memiliki banyak operasi yang ada disetiap classnya.

3.3 Activity Diagram

Pada activity diagram menggambarkan bagaimana aktivitas dari sistem aplikasi yang akan berjalan.

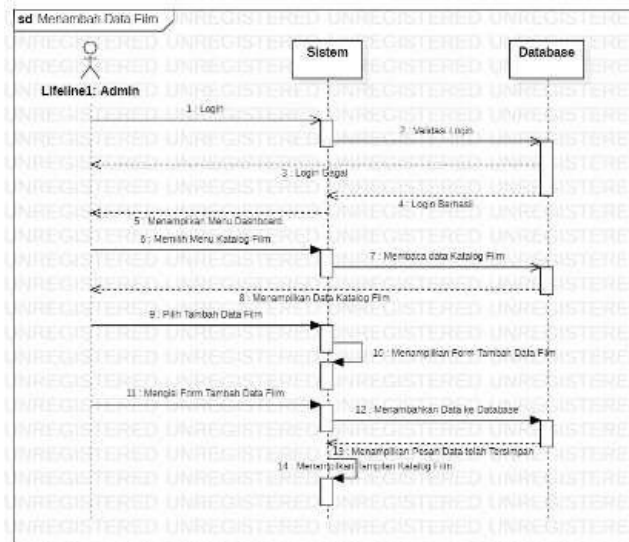


Gambar 4. Activity Diagram Admin Cinema Catalog Movie

Pada **Gambar 4**, terdapat sebuah aktivitas kegiatan yang menggambarkan pengguna Admin yang akan mengakses sebuah sistem tersebut. Beberapa aktivitas tersebut seperti menambahkan, menghapus, dan mengubah data dari data film, kategori, dan user.

3.4 Sequence Diagram

Pada sequence diagram merupakan gambaran dari interaksi aktivitas yang dilakukan oleh sejumlah object dalam urutan waktu.

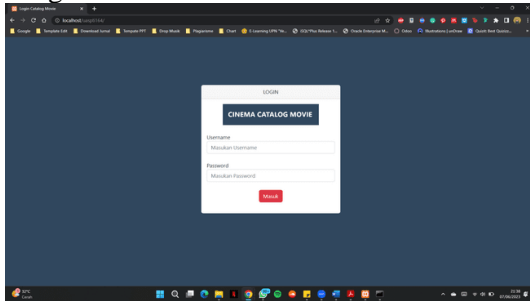


Gambar 5. Use Case Diagram Cinema Catalog Movie

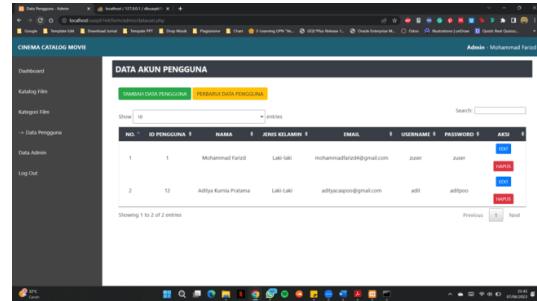
Pada **Gambar 5** tersebut, menggambarkan sebuah interaksi dari beberapa object yang melakukan kegiatan Menambahkan Data Film. Dimana terdapat object Admin, Sistem, dan Database, setiap object tersebut memiliki tugas atau interaksi yang berbeda beda pada setiap object tersebut.

3.5 Implementasi

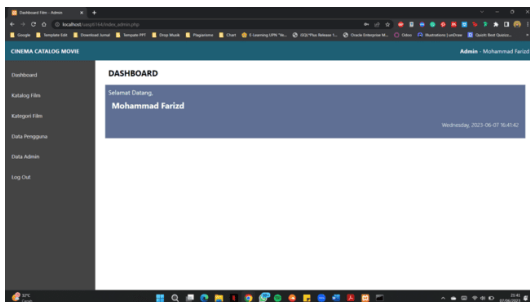
Pada tahapan implementasi ini merupakan hasil tampilan dari desain system yang telah dibuat sekaligus dijelaskan di tahapan selanjutnya. Berikut merupakan hasil tampilan dari penelitian yang telah dibuat sekaligus dirancang mengenai sistem informasi Cinema Catalog Movie berbasis website dan desktop dengan menggunakan hak akses admin, dapat dilihat pada **Gambar 6** sampai dengan **Gambar 18**.



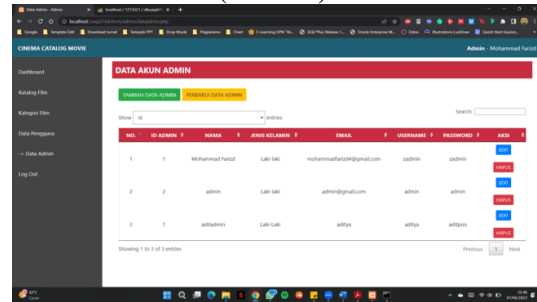
Gambar 6. Halaman Login (Website)



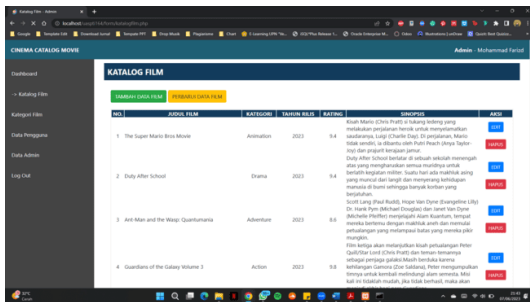
Gambar 10. Halaman Akun Pengguna (Website)



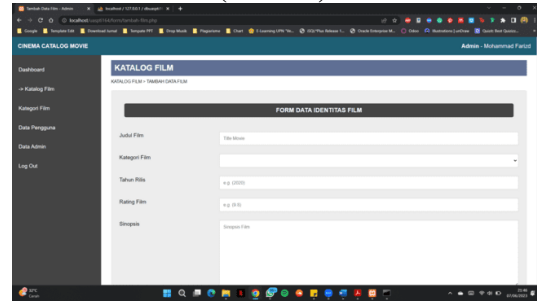
Gambar 7. Halaman Dashboard (Website)



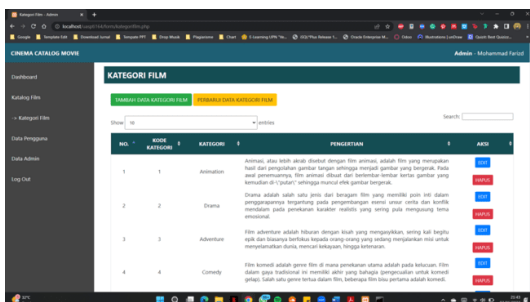
Gambar 11. Halaman Akun Admin (Website)



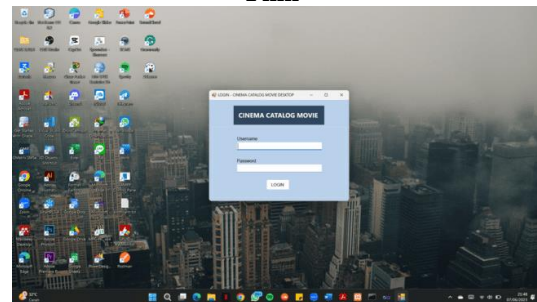
Gambar 8. Halaman Katalog Film (Website)



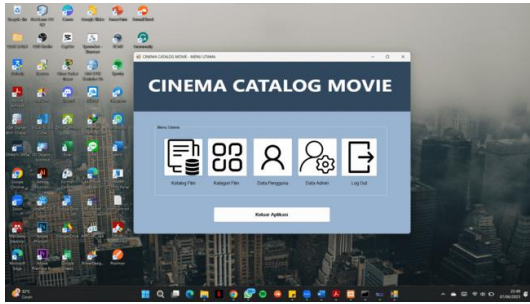
Gambar 12. Halaman Form Tambah Data Film



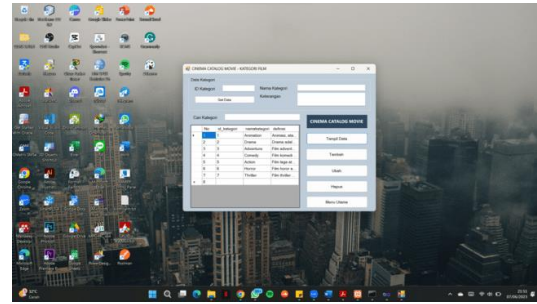
Gambar 9. Halaman Kategori Film (Website)



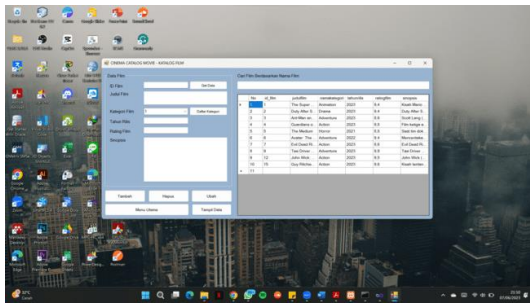
Gambar 13. Tampilan Login (Desktop)



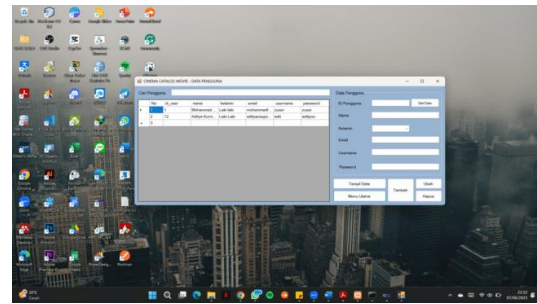
Gambar 14. Tampilan Menu Utama (Desktop)



Gambar 16. Tampilan Menu Kategori Film (Desktop)



Gambar 15. Tampilan Menu Katalog Film (Desktop)



Gambar 17. Tampilan Data Pengguna (Desktop)

3.6 Pengujian

Tahapan pengujian sistem ini dilaksanakan untuk mengetahui bagaimana respon dari sistem yang telah dibuat berjalan dengan baik dan berfungsi. Metode pengujian ini dilakukan dengan metode blackbox testing yang akan memerikan jalannya sistem apakah telah sesuai atau tidak. Berikut merupakan table blackbox testing yang berfokus pada fungsional sistem.

Tabel 1. Hasil Pengujian Black Box Testing Halaman Data Film (Desktop dan Website)

Form Uji	Skenario Pengujian	Hasil Yang Diharapkan	Hasil Pengujian
Tambah Data Katalog Film	Menekan pilihan aksi “Tambah Data Film”	Sistem akan menampilkan form tambah data film.	Sukses
	Menekan pilihan aksi “Simpan” pada form Katalog Film	Muncul pesan “Data Film Baru Berhasil Di Simpan!” dan menampilkan halaman Katalog Film	Sukses
Ubah Data Katalog Film	Menekan pilihan aksi “Edit” pada table Katalog Film	Sistem akan menampilkan form ubah pada Identitas Film	Sukses
	Menekan pilihan aksi “Simpan” pada form Data Identitas Film	Muncul pesan “Data Film Berhasil Di Ubah!” dan menampilkan halaman Katalog Film	Sukses
Cari Data Katalog Film	Melakukan pencarian pada text box/search box	Sistem akan menampilkan kata yang dicari pada table Katalog Film	Sukses

Hapus Data Katalog Film	Menekan pilihan aksi “Hapus” pada table Katalog Film	Sistem akan menghapus dan muncul pesan “Data Film Berhasil Di Hapus!”	Sukses
-------------------------	--	---	--------

3.7 Pemeliharaan

Dalam tahapan pemeliharaan harus selalu memastikan bahwa sistem yang telah dirancang dan dibuat selalu berfungsi dengan baik, serta memperhatikan sekaligus pengecekan secara berkala terhadap proses data yang berjalan. Sehingga dapat menghasilkan luaran sistem informasi yang sesuai dengan harapan.

4. KESIMPULAN DAN SARAN

Dalam pembuatan sistem yang berbasis Desktop dan Web Client Penelitian ini menghasilkan sebuah sistem informasi yang disebut Cinema Catalog Movie. Sistem ini dirancang untuk mengelola dan menyajikan informasi mengenai katalog film. Menggunakan beberapa diagram untuk menggambarkan dan mengorganisir fungsionalitas sistem. Diagram yang digunakan antara lain Use Case Diagram, Class Diagram, Activity Diagram, dan Sequence Diagram.

Berdasarkan hasil yang telah disampaikan peneliti, maka dapat disimpulkan dengan penerapan REST API menggunakan Retrofit pada sistem Informasi film Dengan demikian, penelitian ini berhasil menghasilkan sebuah sistem informasi Cinema Catalog Movie yang memiliki fungsionalitas yang jelas dan telah diimplementasikan dengan baik dalam bentuk tampilan website dan desktop. Tahapan pengujian dan pemeliharaan juga dilakukan untuk memastikan keberhasilan dan kinerja sistem secara keseluruhan.

5. DAFTAR RUJUKAN

- [1] S. N. Yanti and E. Rihyanti, “Penerapan Rest API untuk Sistem Informasi Film Secara Daring,” vol. 6, no. 1, pp. 195–201, 2021.
- [2] R. Affriantari, “PERANCANGAN FITUR REKOMENDASI FILM DI WEBSITE SOLO MOVIE DENGAN MENGGUNAKAN AFFRIANTARI ROCHMAH I 0306013 JURUSAN TEKNIK INDUSTRI FAKULTAS TEKNIK,” 2010.
- [3] R. Christover, “Penerapan Rest API Menggunakan Retrofit Untuk Sistem Informasi Film Berbasis Android (Studi Kasus : Sinopsis Film),” vol. 3, no. 2, pp. 159–170, 2022.
- [4] R. Akram, “Sistem Informasi Terintegrasi Untuk Manajemen Persediaan Dan Distribusi Barang Pdam Kota Langsa,” vol. 12, pp. 33–41, 2023.
- [5] W. G. Wardhana, I. Arwani, and B. Rahayudi, “Implementasi Teknologi Restful Web Service Dalam Pengembangan Sistem Informasi Perekaman Prestasi Mahasiswa Berbasis Website (Studi Kasus : Fakultas Teknologi Pertanian Universitas Brawijaya),” vol. 4, no. 2, pp. 680–689, 2020.
- [6] S. Informasi, P. Ayam, and B. Berbasis, “Penerapan RESTful Web Service dan JSON pada Application Programming Interface (API) Sistem Informasi Perkembangan Ayam Broiler Berbasis Kemitraan Artikel Ilmiah Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Kristen Satya Wacana Salatiga Januari 2015,” 2015.
- [7] N. W. Wardani, P. Gede, S. Cipta, and G. S. Mahendra, “Sistem Informasi Laporan Keuangan pada Salon Berbasis Website Dengan Metode SDLC,” no. December, 2021, doi: 10.33173/jsikti.118.
- [8] W. S. Dharmawan, D. Purwaningtias, and D. Risdiansyah, “Penerapan Metode SDLC Waterfall Dalam Perancangan Sistem Informasi Administrasi Keuangan Berbasis Desktop,” vol. VI, no. 2, 2018.
- [9] R. Choirudin, A. Adil, P. Studi, and T. Informatika, “IMPLEMENTASI REST API WEB SERVICE DALAM MEMBANGUN APLIKASI,” vol. 18, no. 2, pp. 284–293, 2019.

- [10] Decoding Intern, “Apa itu Web Service? Beserta Pengertian dan Contohnya - Dicoding Blog,” *Dicoding Blog*, Jan. 12, 2021. <https://www.dicoding.com/blog/apa-itu-web-service/>
- [11] Amarizky Yoga Pratama, & Jeffri Alfa Razaq. (2023). Integrasi Sistem Informasi Akademik Dan Elearning Moodle Dengan Rest Api. *Jurnal Manajemen Informatika Dan Sistem Informasi*, 6(1), 26–38. <https://doi.org/10.36595/misi.v6i1.696>