

IMPLEMENTASI ALGORITMA ENKRIPSI PADA SISTEM INFORMASI MANAJEMEN HELPDESK BERBASIS DESKTOP

Royan Habibie Sukarna^{1*}, Andi Moch Januriana², Holilah³, Mohamad Hilman⁴
^{1,2,3,4}Program Studi Informatika, Fakultas Teknik, Universitas Sultan Ageng Tirtayasa

Article Info

Article history:

Received September 25, 2023
Accepted November 30, 2023
Published November 30, 2023

Keywords:

Helpdesk System, Extreme Programming, AES256

Helpdesk System, Extreme Programming, AES256

ABSTRACT

Paper ini membahas implementasi algoritma enkripsi pada Sistem Informasi Manajemen *Helpdesk* berbasis desktop yang dikembangkan khusus untuk tim IT di rumah sakit swasta. Tim IT dalam rumah sakit swasta sering menghadapi tantangan yang kompleks karena luasnya area rumah sakit dan sumber daya yang terbatas. Pembuatan aplikasi *Helpdesk* menggunakan Delphi dan *database MySQL* menjadi sangat penting untuk memberikan dukungan teknis yang efisien. Salah satu aspek kunci dalam aplikasi ini adalah fitur *Chat* interaktif yang dibangun menggunakan sistem socket *server-client*. Dalam konteks komunikasi ini, *algoritma* enkripsi diperlukan untuk memastikan keamanan data yang ditransmisikan antara tim IT dan pengguna sistem. Ini membantu melindungi data sensitif pasien dan informasi teknis dari potensi risiko kebocoran. Ruang lingkup aplikasi *Helpdesk* ini mencakup manajemen tiket (*ticketing*), pemeriksaan rutin (*routine check*), pemeliharaan (*maintenance*), rotasi alat, serta berbagai dukungan lainnya. Implementasi algoritma enkripsi dalam aplikasi ini secara efektif meningkatkan tingkat keamanan data dalam seluruh proses operasional. Aplikasi *helpdesk* dibangun menggunakan metode *Extreme Programming (XP)*.

This paper discusses the implementation of encryption algorithms in a desktop-based Helpdesk Management Information System developed specifically for IT teams in private hospitals. IT teams in private hospitals often face complex challenges due to the large hospital area and limited resources. Creating a Helpdesk application using Delphi and a MySQL database is very important to provide efficient technical support. One of the key aspects of this application is the interactive chat feature, which is built using a server-client socket system. In this communication context, encryption algorithms are needed to ensure the security of data transmitted between the IT team and system users. This helps protect sensitive patient data and technical information from the potential risk of leakage. The scope of this Helpdesk application includes ticket management, routine checks, maintenance, tool rotation, and various other support. The implementation of encryption algorithms in this application effectively increases the level of data security in all operational processes. The helpdesk application was built using the Extreme Programming (XP) method.



Corresponding Author:

Royan Habibie Sukarna*,
Program Studi Informatika, Universitas Sultan Ageng Tirtayasa
Email: *royan@untirta.ac.id

1. PENDAHULUAN

Dalam era teknologi informasi yang semakin maju, sistem informasi manajemen Helpdesk menjadi salah satu komponen yang sangat penting dalam operasional rumah sakit swasta. Tim IT di rumah sakit swasta seringkali menghadapi tantangan yang signifikan dalam menjaga kelancaran sistem teknologi informasi yang mendukung berbagai aspek perawatan pasien, administrasi, dan manajemen. Luasnya area rumah sakit dan keterbatasan sumber daya manusia membuat peran sistem Helpdesk menjadi tak tergantikan dalam menyediakan dukungan teknis yang efisien.

Dalam upaya untuk meningkatkan efektivitas dan keamanan layanan yang diberikan oleh tim IT, pembuatan aplikasi Helpdesk berbasis desktop telah menjadi prioritas. Aplikasi ini dikembangkan menggunakan Delphi dan didukung oleh database MySQL yang dapat mengintegrasikan berbagai fungsi penting seperti manajemen tiket (ticketing), pemeriksaan rutin (routine check), pemeliharaan (maintenance), rotasi alat, serta berbagai dukungan lainnya.

Salah satu aspek terpenting dalam pengembangan aplikasi Helpdesk ini adalah keamanan data. Dengan kemunculan teknologi komunikasi yang semakin canggih, interaksi antara tim IT dan pengguna sistem menjadi semakin penting. Fitur Chat interaktif yang dibangun menggunakan sistem socket server-client memungkinkan komunikasi real-time yang memadukan kenyamanan dan efisiensi. Namun, keamanan data dalam komunikasi ini harus dijamin secara ketat. Oleh karena itu, implementasi algoritma enkripsi menjadi sebuah keharusan untuk melindungi data sensitif yang ditransmisikan dalam fitur Chat dan seluruh aplikasi Helpdesk.

Penelitian ini bertujuan untuk mendeskripsikan langkah-langkah implementasi algoritma enkripsi dalam konteks aplikasi Helpdesk ini. Kami juga akan mengkaji dampak implementasi ini terhadap tingkat keamanan data dan efisiensi operasional tim IT. Hasil penelitian ini diharapkan akan memberikan landasan yang kuat untuk meningkatkan keamanan dan efektivitas dalam pelayanan rumah sakit swasta.

Meningkatkan dan menjamin ketersediaan layanan teknologi informasi (TI) yang berkualitas tinggi diperlukan agar kerangka kerja Manajemen Layanan Teknologi Informasi (ITSM) dapat memenuhi kebutuhan organisasi dan mampu menyediakan fungsi manajemen pengetahuan, perlu dilakukan analisis menyeluruh dari perencanaan hingga pelaksanaan layanan Helpdesk Sistem Informasi [1]. Adapun teknik yang digunakan untuk melakukan pengujian sistem dapat menggunakan BlackBox [2]. Helpdesk bertujuan agar proses pemantauan dan laporan tentang komplain yang disampaikan, serta permintaan akses dan kendala perangkat komputer dan jaringan, dapat dilacak dan dievaluasi dengan baik untuk menemukan solusi yang tepat [3].

2. METODE PENELITIAN

2.1 AES (Advanced Encryption Standard)

AES adalah jenis algoritma kriptografi yang memiliki karakteristik simetri dan cipher block, sehingga menggunakan kunci yang sama saat enkripsi dan dekripsi dan menghasilkan blok dengan jumlah bit yang spesifik [4]. Input dan output algoritma AES terdiri dari urutan data 128 bit [5]. Urutan data ini juga disebut blok data atau plaintext, yang akan dienkripsi menjadi ciphertext di masa mendatang. Kunci enkripsi AES terdiri dari kunci dengan panjang 128 bit, 192 bit, atau 256 bit. Jumlah putaran yang akan digunakan pada algoritma AES ini akan dipengaruhi oleh perbedaan panjang kunci.

2.2 Extreme Programming (XP)

Extreme Programming (XP) adalah pengembangan rekayasa perangkat lunak yang sarannya adalah tim skala kecil hingga medium. Metode ini juga dapat digunakan untuk pengembangan sistem dengan persyaratan yang tidak jelas atau perubahan yang sangat cepat [6]. XP adalah metode yang sangat responsif terhadap perubahan. Kelebihan lain dari metode ini adalah bahwa ia memberikan tahapan dalam waktu pengerjaan yang singkat sesuai dengan fokus yang akan dicapai. Metode *Ekstrim Programming* (XP), juga dikenal sebagai metode XP, adalah jenis model pengembangan perangkat lunak yang memiliki tahapan pengembangan sistem yang menjadi lebih efisien, adaptif, dan fleksibel. XP tidak hanya berfokus pada pemrograman, tetapi mencakup sebagian besar area pengembangan perangkat lunak. Tahapan metode pengembangan sistem *Extreme Programming* [7] adalah sebagai berikut:

a. *Planning* (Perencanaan)

Dalam tahap awal pembangunan sistem, beberapa kegiatan perencanaan dilakukan, seperti mengidentifikasi masalah, mengevaluasi kebutuhan, dan menetapkan jadwal pelaksanaan pembangunan sistem. Tahap perencanaan dapat dimulai dengan mendengarkan kumpulan kebutuhan aktifitas sistem agar pengguna dapat memahami proses bisnis untuk sistem dan mendapatkan gambaran yang jelas dalam mengenai fitur utama, fungsionalitas dan keluaran yang diinginkan.

b. *Design* (Perancangan)

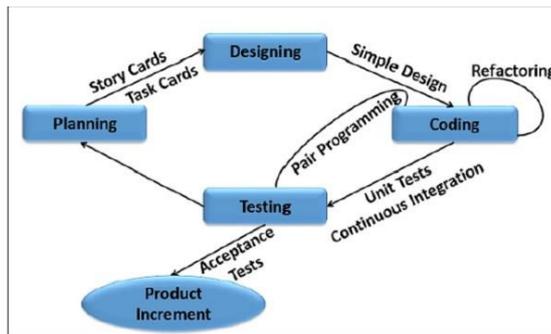
Pada tahap selanjutnya, perancangan, kegiatan pemodelan dilakukan, termasuk pemodelan sistem, arsitektur, dan basis data. Pemodelan sistem dan arsitektur menggunakan diagram aliran (*flowchart*), sedangkan pemodelan basis data menggunakan *Entity Relationship Diagram* (ERD).

c. *Coding* (Pengkodean)

Ini adalah proses penerapan pemodelan yang sudah dibuat ke dalam bentuk antarmuka pengguna menggunakan bahasa pemrograman Delphi dengan metode terstruktur. Piranti lunak MySQL digunakan untuk mengelola sistem basis data.

d. *Testing* (Pengujian)

Setelah tahapan pengkodean selesai, tahapan pengujian sistem dilakukan untuk mengidentifikasi kesalahan apa pun yang terjadi saat aplikasi berjalan dan untuk memastikan apakah sistem yang dibangun sudah memenuhi kebutuhan pengguna. Pada tahap ini, metode *blackbox testing* digunakan untuk menguji bentuk berbeda untuk melihat apakah sudah berjalan sesuai dengan fungsinya masing-masing.



Gambar 1. Metode Extreme Programming

3. HASIL DAN PEMBAHASAN

3.1. Enkripsi AES

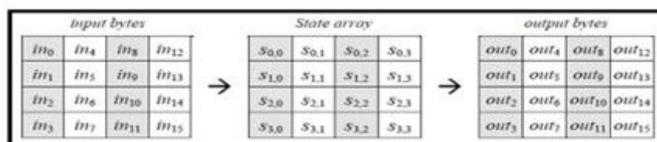
Penelitian ini menggunakan Algoritma AES-256 sebagai standar enkripsi untuk mengumpulkan data teks dan pesan dari fitur chat interaktif. Proses enkripsi dilakukan pada setiap blok penyandian, satu blok per blok, hingga blok terakhir. Tidak ada padding seperti yang dilakukan oleh beberapa algoritma kriptografi lainnya jika ukuran blok terakhir lebih kecil dari blok penyandian yang telah ditentukan.

Untuk menggantikan DES (*Data Encryption Standard*), AES adalah simetri *block cipher*. *Input* dan *output* algoritma AES terdiri dari rangkaian data 128 bit. Rangkaian data ini disebut *block data* atau *plaintext*, dan mereka akan dienkripsi menjadi ciphertext di masa mendatang. AES-128, AES-192, dan AES-256 adalah tiga jenis kunci algoritma AES, yang merupakan algoritma simetris yang menggunakan kunci yang sama untuk baik enkripsi maupun dekripsi.

Untuk setiap proses putaran, masing-masing tipe menggunakan kunci internal bulat. Proses enkripsi AES-128 diputar sepuluh kali ($a=10$), yang menunjukkan hasil berikut:

- Addroundkey.
- Putaran sebanyak $a-1$ kali, proses yang dilakukan pada setiap putaran adalah: *SubBytes*, *ShiftRows*, *MixColumns*, dan *AddRoundKey*.
- Final round, adalah proses untuk putaran terakhir yang meliputi *SubBytes*, *ShiftRows*, dan *AddRoundKey*. Sedangkan pada proses dekripsi AES-128, proses putaran juga dikerjakan sebanyak 10 kali ($a=10$).

Pada dasarnya, AES digunakan terhadap array byte dua dimensi yang disebut *state*. *State* memiliki $NROWS \times NCOLS$. Data input $in_0, in_2, in_3, in_4, in_5, in_6, in_7, in_8, in_9, in_{10}, in_{11}, in_{12}, in_{13}, in_{14}$, dan in_{15} disalin ke dalam *array state* saat proses enkripsi dan dekripsi dimulai. Setelah itu, keluarannya akan dimasukkan ke dalam *array keluar*. Gambar 1 menunjukkan proses penyalinan dari *byte input*, *byte state array*, dan *byte output*:



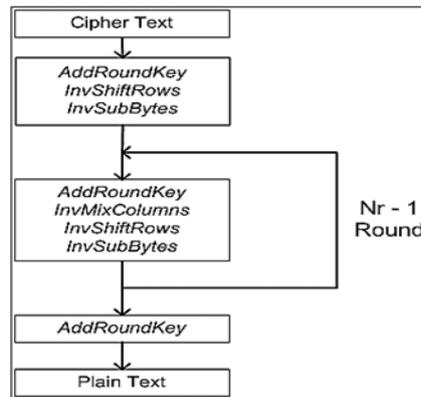
Gambar 2. Proses Input Bytes, State Array, dan Output Bytes

Pada proses enkripsi [8], empat transformasi berbeda digunakan dalam algoritma AES, yaitu:

- SubBytes*()
- ShiftRows*()
- MixColumns*()
- AddRoundKey*()

2.3 Dekripsi AES

Untuk algoritma AES, invers cipher dapat dibalik dan diterapkan dalam arah yang berlawanan untuk membuat invers cipher yang mudah dipahami [5]. Beberapa contoh transformasi byte yang digunakan pada invers cipher adalah *InvShiftRows*, *InvSubBytes*, *InvMixColumns*, dan *AddRoundKey*. Skema berikut menunjukkan algoritma dekripsi:



Gambar 3. Ilustrasi Proses Dekripsi AES

2.4 Perencanaan (Planning)

Beberapa analisis kebutuhan ditetapkan untuk sistem perancangan Sistem Informasi Manajemen Helpdesk Berbasis Desktop dengan metode Extreme Programming (XP), termasuk:

a. Kebutuhan Pengguna

Kebutuhan pengguna berikut digunakan oleh pembangunan aplikasi ini:

1) Kebutuhan Tim IT

- Login sebelum masuk ke halaman dashboard dan menu utama.
- Tim IT dapat mengubah, menghapus, data pemeliharaan dan pengecekan rutin.
- Tim IT dapat mengubah, menghapus, data pengajuan yang berasal dari staff divisi.
- Tim IT dapat mematikan, menghidupkan fitur *live chat*.
- Data laporan hasil kerja dapat dibuat dan dipantau oleh tim IT.

2) Kebutuhan Staff Divisi

- Staff divisi lain dapat melihat status *request*, dan menambahkan data request baru.
- Staff divisi lain dapat memantau status personil Tim IT.
- Staff divisi lain dapat menggunakan fitur *live chat*.

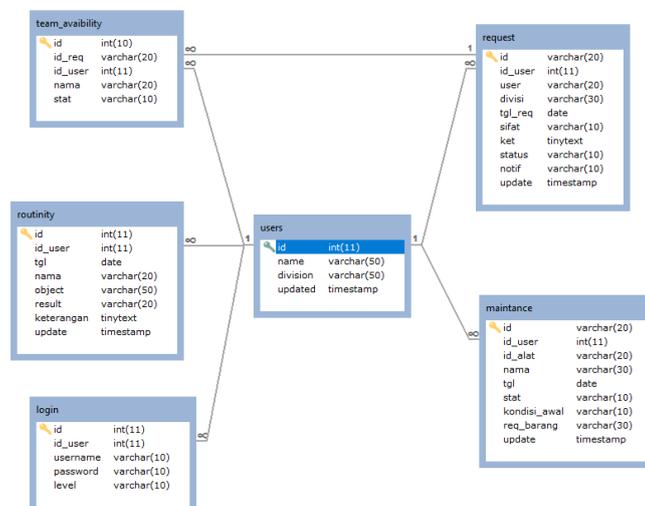
b. Kebutuhan Sistem

- Sistem memberikan hak akses untuk staff divisi membuat ticket baru.
- Sistem akan menampilkan data ticketing pada tim IT sesuai dengan prioritas masalah.
- Sistem akan melakukan enkripsi pesan pada fitur *live chat*.
- Apabila staff melakukan proses logout, sistem akan memberhentikan akses pelayanan.

2.5 Perencanaan (Design)

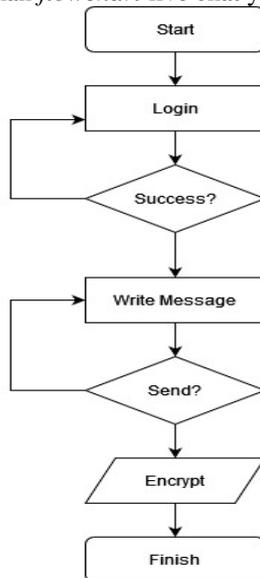
2.5.1 ERD (Entity Relationship Diagram)

Model Entity-Relationship Diagram (ERD) adalah alat penting dalam dunia pemodelan basis data yang digunakan untuk merancang dan menggambarkan hubungan antara entitas atau objek dalam sistem informasi. Gambar 4 menunjukkan hubungan dari tiap entitas.



Gambar 4. Entity Relationship Diagram

Tahap perancangan adalah tahapan dimana alur logic dari sistem dibuat. Alur ini digambarkan dalam diagram alir (Flowchart). Gambar 5 menunjukkan *flowchart* live chat yang ada di aplikasi helpdesk.

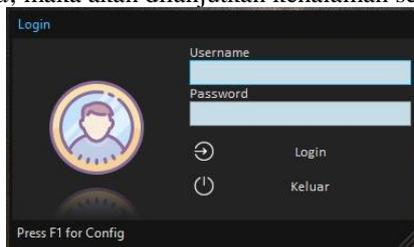


Gambar 5. *Flowchart Live Chat*

2.6 Pengkodean (Coding)

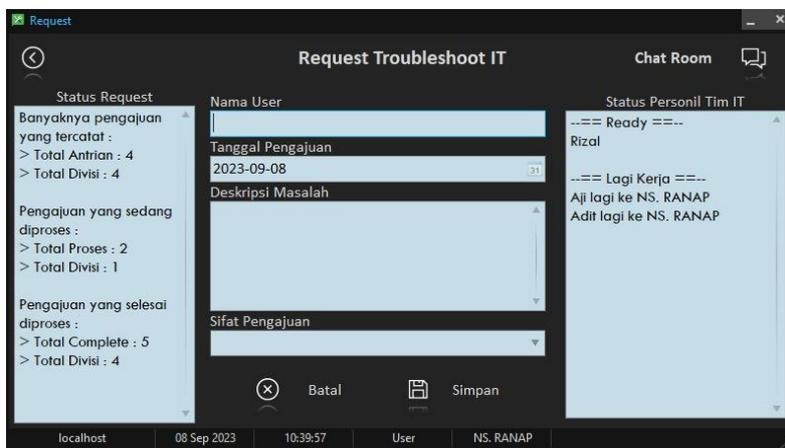
Setelah itu, rancangan dimasukkan ke dalam kode program, yang menghasilkan prototype program sebagai berikut:

Tampilan halaman login untuk Tim IT dan Staff Divisi digunakan untuk otentikasi dan otorisasi. Setelah memasukkan *username* dan *password*, maka akan dilanjutkan kehalaman sesuai dengan level otorisasi.



Gambar 6. *Form Login*

Gambar 7 merupakan tampilan dari *form ticketing user*. Form ini digunakan oleh staff dari divisi rumah sakit untuk menambahkan keluhan yang membutuhkan bantuan dari tim IT. Form ini pun menampilkan informasi status *request* yang dikelola oleh aplikasi dan status ketersediaan personil tim IT.



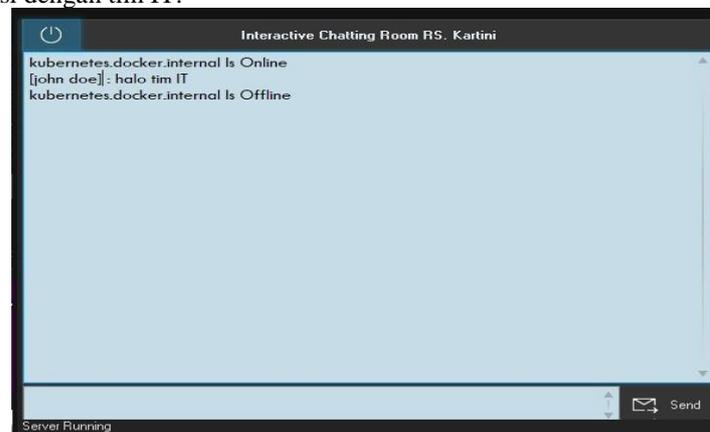
Gambar 7. *Form Ticketing User*

Gambar 8 merupakan tampilan dari fitur *live chat* dari sisi pengguna staff divisi. Staff divisi dapat melihat apakah tim IT sedang *online* atau *offline*. Form ini dapat digunakan untuk melakukan percakapan antar divisi dengan tim IT.



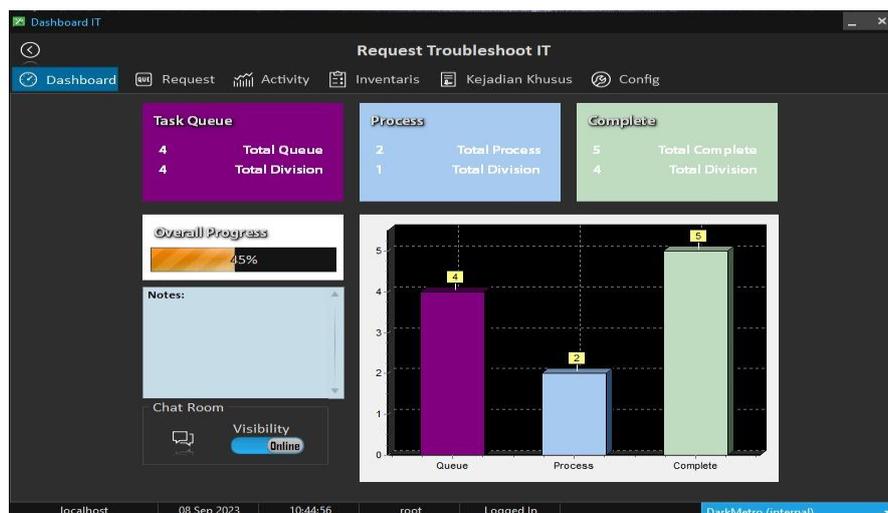
Gambar 8. Form Live Chat User

Gambar 9 merupakan tampilan dari fitur *live chat* dari sisi pengguna tim IT. Tim IT dapat melihat percakapan dari berbagai divisi yang sedang *online* atau *offline*. Form ini dapat digunakan untuk melakukan percakapan antar divisi dengan tim IT.



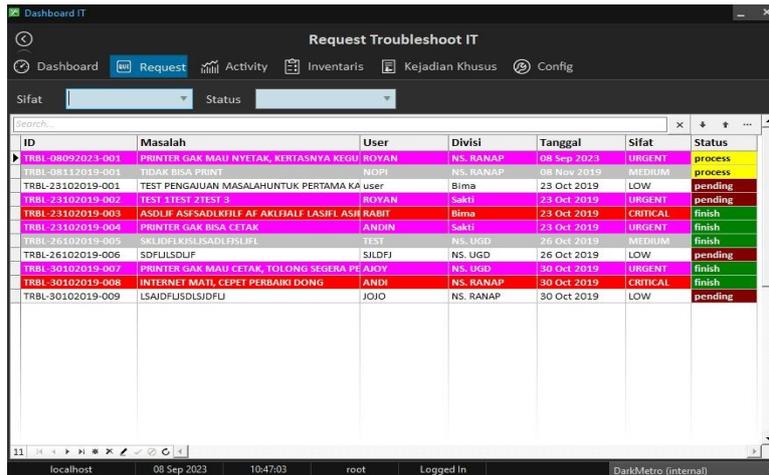
Gambar 9. Form Live Chat

Gambar 10 adalah tampilan *dashboard* yang berisikan rekap informasi dari pengajuan *request* seluruh divisi yang sudah dikerjakan, sedang diproses, dan belum diproses.



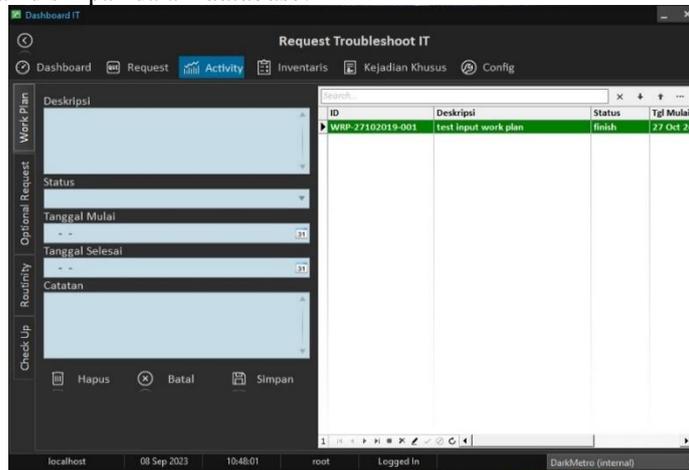
Gambar 10. Form Dashboard

Gambar 11 adalah tampilan *form request*. Form ini menampilkan pengajuan permintaan perbaikan ataupun pemeriksaan dan kendala dari seluruh divisi. Status *urgency* direpresentasikan oleh masing-masing warna.



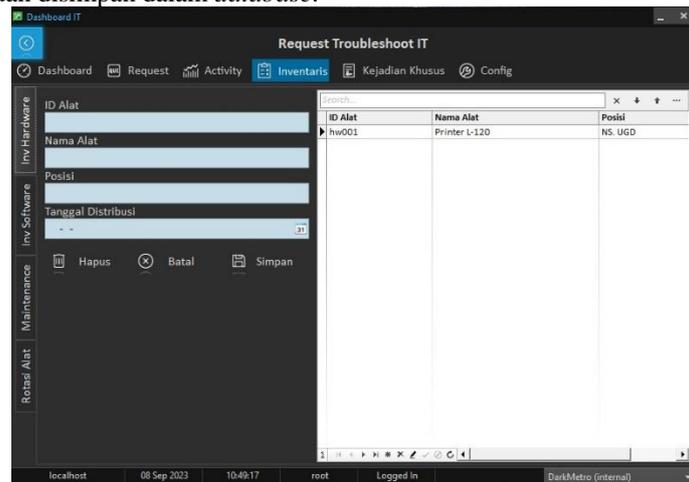
Gambar 11. Form Request

Gambar 12 adalah tampilan dari form routine. Form ini digunakan oleh Tim IT untuk melakukan pemeriksaan berkala dan disimpan dalam database.



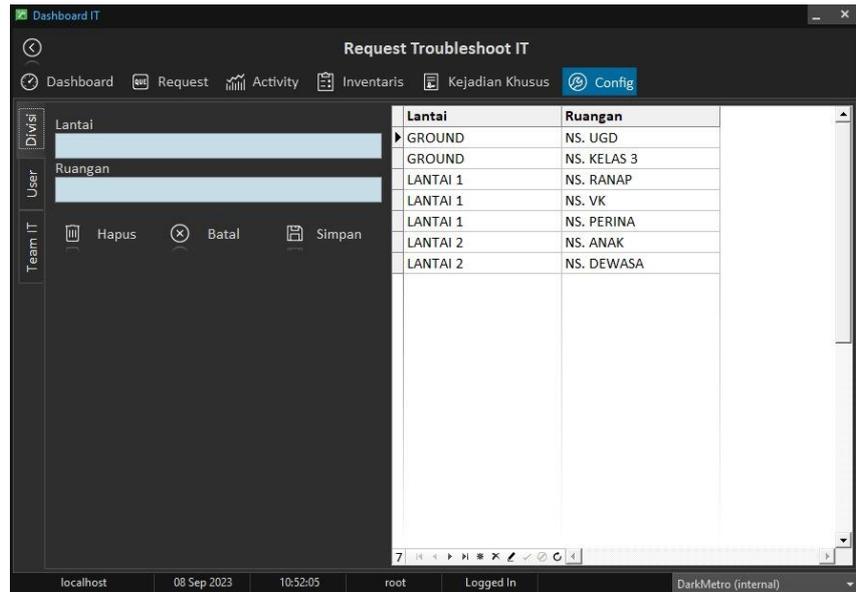
Gambar 12. Form Routine

Gambar 13 adalah tampilan dari form maintenance. Form ini digunakan oleh Tim IT untuk melakukan pemeliharaan berkala dan disimpan dalam database.



Gambar 13. Form Maintenance

Gambar 14 adalah tampilan pengaturan untuk mengatur divisi yang ada di rumah sakit. Penambahan, dan pengurangan divisi dapat dilakukan pada form ini. Serta konfigurasi koneksi dan manajemen user dilakukan pada form ini.



Gambar 14. Form Config Divisi

2.7 Pengujian (Testing)

Setelah itu, sistem informasi manajemen *helpdesk* diuji dengan *BlackBox Testing*, yang menguji validasi *input* dan *output*. Tabel 1 menunjukkan hasil pengujian menggunakan *BlackBox Testing*.

Tabel 1. Pengujian *BlackBox*

No	Aktivitas Pengujian	Realisasi yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Klik tombol login pada form login setelah melakukan pengisian <i>username</i> dan <i>password</i> yang benar dan salah.	Muncul pesan sukses “login sukses” jika <i>username</i> dan <i>password</i> yang dimasukkan benar, dan pesan “user atau <i>password</i> salah” jika data yang dimasukkan salah.	Pesan yang muncul sesuai dengan data yang dimasukkan	[x] Diterima [] Ditolak
2	Klik tombol Simpan pada form ticketing user setelah mengisi data dengan benar	Data tersimpan, kolom isian akan dibersihkan, dan merubah tampilan status request	Kolom isian menjadi bersih kembali, dan tampilan status request berubah menjadi bertambah	[x] Diterima [] Ditolak
3	Klik tombol sign in dan tombol kirim pada form live chat user	Muncul pesan bahwa user online dan pesan yang dikirim tampil pada layar percakapan	Pesan online muncul setelah tombol sign in ditekan, dan pesan percakapan muncul pada layar	[x] Diterima [] Ditolak
4	Klik tombol send pada form live chat	Pesan yang dikirim tampil pada layar percakapan dan terkirim ke semua client	Pesan yang dikirim oleh server tampil pada seluruh client	[x] Diterima [] Ditolak
5	Membuka form dashboard untuk memunculkan jumlah statistik request dari user	Form dashboard dapat menampilkan statistik request dari user	Form dashboard berfungsi dengan baik saat ada penambahan request baru	[x] Diterima [] Ditolak
6	Mencoba filter pada form request	Saat memfilter data berdasarkan sifat dan status, maka data yang ditampilkan akan berubah sesuai kondisi	Data yang ditampilkan sesuai dengan kondisi saat filter dijalankan	[x] Diterima [] Ditolak

7	Klik tombol hapus, tombol batal, dan tombol simpan pada form routine, form maintenance, dan form config divisi	Data yang dipilih untuk dihapus akan terhapus, form akan ter-reset jika menekan tombol batal, dan data yang baru akan tampil jika tombol simpan ditekan	Semua tombol berfungsi sesuai dengan kriteria yang disebutkan	[x] Diterima [] Ditolak
---	----------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------	-----------------------------

4. KESIMPULAN

Penggunaan metode *extreme programming* dapat membantu dalam membangun sistem informasi manajemen *helpdesk* berbasis desktop. Penggunaan algoritma enkripsi sangat penting untuk sistem manajemen *Helpdesk* berbasis *desktop* yang menggunakan fitur *live chat*. Aplikasi *Helpdesk* yang berhasil sangat relevan dengan masalah yang dihadapi tim IT dalam mengelola sistem teknologi informasi di lingkungan rumah sakit yang luas dengan sumber daya terbatas.

UCAPAN TERIMA KASIH

Kami ingin mengucapkan terimakasih yang sebesar-besarnya atas dukungan dan kontribusi yang berharga dalam penyelesaian paper ini. Penyusunan paper ini tidak mungkin terwujud tanpa bantuan berbagai pihak yang telah turut serta aktif dalam proses pengembangan sistem ini.

Pertama-tama, kami mengucapkan terimakasih kepada seluruh tim IT di rumah sakit swasta yang telah memberikan wawasan mendalam tentang tantangan yang mereka hadapi sehari-hari. Dukungan teknis dan pemahaman mendalam mengenai kebutuhan mereka telah menjadi landasan utama dalam perancangan dan implementasi aplikasi *Helpdesk* ini.

Terimakasih kepada semua pihak yang secara langsung maupun tidak langsung turut berperan dalam kesuksesan penelitian ini. Semua sumbangan, baik itu dalam bentuk waktu, pengetahuan, atau dukungan moral, sangat berarti bagi kemajuan penelitian ini.

Semoga paper ini dapat memberikan kontribusi positif dalam pengembangan sistem informasi manajemen *Helpdesk* di lingkungan rumah sakit swasta. Kami berharap dapat terus berkolaborasi dan berbagi pengetahuan untuk meningkatkan efisiensi dan keamanan layanan teknis di dunia kesehatan.

REFERENSI

- [1] E. M. Sipayung, C. Fiarni, and E. Aditya, "Perancangan Sistem Informasi Helpdesk Menggunakan Framework ITIL V3," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 6, no. 2, 2017, doi: 10.22146/jnteti.v6i2.308.
- [2] M. Saepuloh, "Perancangan Sistem Informasi Manajemen Helpdesk Berbasis Web dengan Framework Codeigniter dan Mysql," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 8, no. 4, pp. 2261–2276, 2021, doi: 10.35957/jatisi.v8i4.1245.
- [3] Y. Liharja, A. O. Sari, and A. Satriansyah, "Rancang Bangun Sistem Informasi Helpdesk IT Support Berbasis Website," *Infotek J. Inform. dan Teknol.*, vol. 5, no. 1, pp. 157–166, 2022, doi: 10.29408/jit.v5i1.4738.
- [4] R. Tullah, M. I. Dzulhaq, and Y. Setiawan, "Perancangan Aplikasi Kriptografi File Dengan Metode Algoritma Advanced Encryption Standard (AES)," *J. SISFOTEK Glob.*, vol. 6, no. 2, pp. 24–30, 2016.
- [5] V. Yuniati, G. Indriyanta, and A. R. C, "Enkripsi dan dekripsi dengan algoritma aes 256 untuk semua jenis file," *J. Inform.*, vol. 5, no. 1, 2009.
- [6] N. A. Septiani and F. Y. Habibie, "Penggunaan Metode Extreme Programming Pada Perancangan Sistem Informasi Pelayanan Publik," *J. Sist. Komput. dan Inform.*, vol. 3, no. 3, p. 341, 2022, doi: 10.30865/json.v3i3.3931.
- [7] I. Carolina and A. Supriyatna, "Penerapan Metode Extreme Programming dalam Perancangan Aplikasi Perhitungan Kuota SKS Mengajar Dosen," *IKRA-ITH Inform. J. Komput. dan Inform.*, vol. 3, no. 1, pp. 106–113, 2019, [Online]. Available: <https://journals.upi-yai.ac.id/index.php/ikraith-informatika/article/view/306/198>.
- [8] I. F. A. SIREGAR, H. R. SYAHFITRI, and TOMMY, "APLIKASI PENGAMANAN FILE DENGAN ALGORITMA AES256 DAN SHA1," no. 70, pp. 1–8, 2010.