



**PERANCANGAN SISTEM INFORMASI PERIZINAN KELUAR MASUK
SANTRI PADA ASRAMA PERGURUAN DINIYAH PUTERI PADANG
PANJANG DENGAN MENGGUNAKAN *FRAMEWORK* SPRING BOOT
DAN POSTGRESQL**

تصميم نظام تصاريح دخول وخروج الطلاب في سكن معهد دينية بوتري بادانغ بانجانغ باستخدام إطار
عمل سبرينغ بوت وقاعدة بيانات بوستجرس كيو إل

***DESIGN OF A SANTRI ENTRY AND EXIT PERMIT SYSTEM AT THE
DINIYAH PUTERI PADANG PANJANG BOARDING SCHOOL USING
THE SPRING BOOT FRAMEWORK AND POSTGRESQL***

TUGAS AKHIR

*Diajukan Kepada Program Studi Manajemen Informatika
Untuk Memenuhi Salah Satu Syarat Mencapai Gelar Ahli Madya (A.Md)
Dalam Bidang Manajemen Informatika*

CITRA SATYA SAKINAH

NIM : 2050401008

**PROGRAM STUDI MANAJEMEN INFORMATIKA
FAKULTAS EKONOMI DAN BISNIS ISLAM
UNIVERSITAS ISLAM NEGERI (UIN) MAHMUD YUNUS
BATUSANGKAR
TAHUN 2025 M/1446 H**

SURAT PERNYATAAN KEASLIAN

Saya yang bertanda tangan dibawah ini:

Nama : Citra Satya Sakinah
NIM : 2050401008
Tempat/Tanggal Lahir : Bukittinggi, 3 Mei 2000
Fakultas : Fakultas Ekonomi dan Bisnis Islam
Program Studi : Manajemen Informatika

Dengan ini menyatakan bahwa Tugas Akhir yang berjudul **“Perancangan Sistem Informasi Perizinan Keluar Masuk Santri Pada Asrama Perguruan Diniyyah Puteri Padang Panjang Dengan Menggunakan *Framework Spring Boot Dan Postgresql*”** adalah hasil karya sendiri, bukan plagiat. Kecuali dicantumkan sumbernya.

Apabila dikemudian hari terbukti karya ilmiah ini plagiat, maka bersedia menerima sanksi sesuai dengan ketentuan undang-undang yang berlaku. Demikianlah pernyataan ini saya buat untuk dapat digunakan semestinya.

Batusangkar, Februari 2025

Saya yang menyatakan,



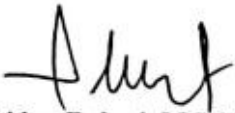
CITRA SATYA SAKINAH
NIM. 2050401008

PERSETUJUAN PEMBIMBING

Pembimbing penulisan Tugas Akhir atas nama Citra Satya Sakinah, NIM: 2050401008, dengan judul "PERANCANGAN SISTEM INFORMASI PERIZINAN KELUAR MASUK SANTRI PADA ASRAMA PERGURUAN DINIYAH PUTERI PADANG PANJANG DENGAN MENGGUNAKAN *FRAMEWORK SPRING BOOT* DAN *POSTGRESQL*" memandang bahwa Tugas Akhir yang bersangkutan telah memenuhi persyaratan ilmiah dan dapat disetujui untuk dilanjutkan ke agenda munaqasah.

Demikian persetujuan ini diberikan untuk dapat digunakan sebagaimana mestinya.

Ketua Jurusan Manajemen
Informatika


Lidya Rahmi, M.Pd.T
NIP. 198805232023212036

Batusangkar, Januari 2025
Pembimbing


Lidya Rahmi, M.Pd.T
NIP. 198805232023212036

Mengetahui,
Dekan Fakultas Ekonomi dan Bisnis Islam
Universitas Islam Negeri (UIN)
Malamud Yunus Batusangkar



Dr. H. Rizal, M.Ag., CRP
NIP. 19731007 200212 1 001

PENGESAHAN TIM PENGUJI

Tugas Akhir atas nama Citra Satya Sakinah, NIM. 2050401008 dengan judul "PERANCANGAN SISTEM INFORMASI PERIZINAN KELUAR MASUK SANTRI PADA ASRAMA PERGURUAN DINIYAH PUTERI PADANG PANJANG DENGAN MENGGUNAKAN *FRAMEWORK* SPRING BOOT DAN *POSTGRES*QL" telah diujikan dalam ujian *Munaqasyah* Fakultas Ekonomi dan Bisnis Islam Universitas Islam Negeri Mahmud Yunus Batusangkar pada hari Rabu tanggal 5 Februari 2025 dan dinyatakan telah lulus serta dapat diterima sebagai salah satu syarat untuk mencapai gelar Ahli Madya Diploma III (A.Md) (D III) dalam Program Studi Manajemen Informatika.

Demikianlah persetujuan ini diberikan untuk dapat digunakan seperlunya.

No	Nama Penguji/NIP	Jabatan	Tanda Tangan	Tanggal
1	Lidya Rahmi, M.Pd.T NIP. 198805232023212036	Ketua Sidang		10/25 /2
2	Dr. Lita Sari Muchlis, M.Kom NIP. 197801222008012017	Penguji 1		10/25 /2
3	Adriyendi, M.Kom NIP. 197701272009121002	Penguji 2		10/25 /2

Batusangkar, Februari 2025

Mengetahui,

Dekan Fakultas Ekonomi dan Bisnis Islam
Universitas Islam Negeri Mahmud Yunus

Batusangkar



Dr. H. Rizal, M.Ag., CRP
NIP. 197310072002121001

ABSTRAK

Citra Satya Sakinah. NIM 2050401008. Judul Tugas Akhir: **“PERANCANGAN SISTEM PERIZINAN KELUAR MASUK SANTRI PADA ASRAMA PERGURUAN DINIYAH PUTERI PADANG PANJANG DENGAN MENGGUNAKAN *FRAMEWORK* SPRING BOOT DAN POSTGRESQL.”** Jurusan Manajemen Informatika Fakultas Ekonomi dan Bisnis Universitas Islam Negeri (UIN) Mahmud Yunus Batusangkar. Dibimbing oleh Ibu Lidya Rahmi, M.Pd.T. Pengelolaan izin keluar masuk santri di Asrama Perguruan Diniyyah Puteri Padang Panjang selama ini masih dilakukan secara manual. Hal ini seringkali menimbulkan berbagai kendala, seperti proses yang lambat, sulitnya mengetahui status izin dengan jelas, dan kesulitan melacak data izin. Untuk mengatasi tantangan tersebut, penelitian ini bertujuan untuk merancang dan membangun sebuah sistem perizinan berbasis web. Sistem ini dikembangkan menggunakan *framework* Spring Boot dan database PostgreSQL untuk mendukung proses yang lebih cepat, praktis, dan terorganisir. Sistem ini dirancang agar santri dapat dengan mudah mengajukan izin secara *online*, sementara pengelola asrama dapat melakukan validasi dan memantau status izin dalam waktu nyata. Selain itu, fitur pelaporan yang tersedia memungkinkan pihak asrama untuk mengakses data perizinan dengan lebih mudah. Pengembangan sistem mengikuti metode *System Development Life Cycle* (SDLC), yang melibatkan perencanaan, analisis, desain, hingga implementasi. Dengan kemudahan yang ditawarkan, sistem ini diharapkan dapat menjadi solusi yang modern dan terpercaya dalam mendukung aktivitas administrasi di asrama.

Kata Kunci: Sistem Perizinan, Digitalisasi, Spring Boot, PostgreSQL, Asrama

ABSTRACT

Citra Satya Sakinah. NIM 2050401008. Final Project Title: "**DESIGN OF A SANTRI ENTRY AND EXIT PERMIT SYSTEM AT THE DINIYAH PUTERI PADANG PANJANG BOARDING SCHOOL USING THE SPRING BOOT FRAMEWORK AND POSTGRESQL**, تصميم نظام تصاريح تصميم نظام تصاريح دخول وخروج الطلاب في سكن معهد دينية بوتري بادانغ بانجانغ باستخدام إطار عمل نظام تصاريح دخول وخروج الطلاب في سكن معهد دينية بوتري بادانغ بانجانغ باستخدام إطار عمل "سبرينغ بوت وقاعدة بيانات بوستجرس كيو إل Department of Information Management, Faculty of Economics and Business, Universitas Islam Negeri (UIN) Mahmud Yunus Batusangkar. Supervised by Mrs. Lidya Rahmi, M.Pd.T. The management of entry and exit permissions for students at the Diniyyah Puteri Padang Panjang Boarding School has so far been carried out manually. This often causes various challenges, such as slow processes, difficulty in clearly identifying permission statuses, and challenges in tracking permission records. To address these issues, this study aims to design and develop a web-based permission system. The system was developed using the Spring Boot framework and PostgreSQL database to support faster, more practical, and organized processes. The system is designed to allow students to easily submit permissions online, while boarding school administrators can validate and monitor permission statuses in real time. Additionally, the available reporting features enable administrators to access permission data more easily. System development followed the System Development Life Cycle (SDLC) method, involving planning, analysis, design, and implementation. With the convenience offered, this system is expected to become a modern and reliable solution to support administrative activities in the boarding school.

Keywords: Permission System, Digitalization, Spring Boot, PostgreSQL, Boarding School.

DAFTAR ISI

HALAMAN JUDUL
SURAT PERNYATAAN KEASLIAN
HALAMAN PERSETUJUAN.....
LEMBAR PENGESAHAN
PERSEMBAHAN.....
ABSTRAK	i
ABSTRACT	ii
KATA PENGANTAR.....	Error! Bookmark not defined.
DAFTAR ISI.....	iii
DAFTAR GAMBAR.....	vi
DAFTAR TABEL.....	vii
BAB I PENDAHULUAN.....	1
A. Latar Belakang	1
B. Identifikasi Masalah.....	3
C. Batasan Masalah.....	3
D. Rumusan Masalah	4
E. Tujuan Penelitian	4
F. Manfaat Penelitian	4
G. Metode Pengembangan Sistem	5
H. Sistematika Penulisan	6
BAB II LANDASAN TEORI.....	9
A. Gambaran Umum Perguruan Diniyyah Puteri Padang Panjang ...	9
1. Sejarah Instansi	9
2. Visi dan Misi.....	9
3. Profil Instansi	10
B. Konsep Dasar Sistem Informasi.....	10

1.	Sistem.....	10
2.	Informasi	13
3.	Sistem Informasi	16
C.	Konsep Perizinan Keluar Masuk Santri di Asrama.....	18
D.	Alat Bantu Perancangan Sistem	21
1.	<i>Unified Modelling Language (UML)</i>	21
E.	Perangkat Lunak Pembangunan Sistem	26
1.	Java.....	26
2.	Spring Boot	27
3.	ReactJS.....	28
4.	PostgreSQL	30
5.	Postman.....	31
6.	Visual Code Studio	32
BAB III	ANALISIS DAN HASIL	33
A.	Perencanaan Sistem.....	33
1.	Identifikasi Kebutuhan.....	33
2.	Penentuan Ruang Lingkup Sistem	33
3.	Sumber Daya yang Diperlukan	34
B.	Analisis Sistem.....	35
1.	Sistem yang Sedang Berjalan.....	35
2.	Sistem yang Diusulkan.....	35
C.	Perancangan Sistem	38
1.	<i>Unified Modelling Language (UML)</i>	38
2.	Antarmuka (<i>Interface</i>).....	45
BAB IV	PENUTUP	59

A. KESIMPULAN	59
B. SARAN	59
DAFTAR PUSTAKA	61
LAMPIRAN.....	Error! Bookmark not defined.
LISTING PROGRAM.....	Error! Bookmark not defined.

DAFTAR GAMBAR

Gambar 2.1 Karakteristik Sistem.....	11
Gambar 2.2 Siklus Informasi.....	16
Gambar 3.1 <i>Use Case Diagram</i>	38
Gambar 3.2 <i>Activity Diagram Login</i>	39
Gambar 3.3 <i>Activity Diagram</i> Pengajuan Izin.....	40
Gambar 3.4 <i>Activity Diagram</i> Pengelolaan Data Pengguna.....	41
Gambar 3.5 <i>Activity Diagram</i> Pengelolaan Data Wali.....	42
Gambar 3.6 <i>Activity Diagram</i> Pengelolaan Data Asrama	43
Gambar 3.7 <i>Class Diagram</i>	44
Gambar 3.8 Tampilan Menu <i>Login</i>	45
Gambar 3.9 Tampilan Pengajuan Permohonan Izin	46
Gambar 3.10 Tampilan Verifikasi Permohonan Izin.....	46
Gambar 3.11 Tampilan Verifikasi Izin Keluar	47
Gambar 3.12 Tampilan Verifikasi Izin Kembali	47
Gambar 3.13 Tampilan Penambahan Data Santri	48
Gambar 3.14 Tampilan Penambahan Data Penjaga	48
Gambar 3.15 Tampilan Penambahan Data Wali	49
Gambar 3.16 Tampilan Penambahan Data Asrama.....	49
Gambar 3.17 Tampilan Pengelolaan Data SantriWali.....	50
Gambar 3.18 Tampilan Riwayat Izin	50
Gambar 3.19 Tampilan Profil Santri	51
Gambar 3.20 Tampilan Profil Penjaga	51
Gambar 3.21 Tampilan Profil Admin.....	52
Gambar 3.22 Tampilan Profil Wali	52
Gambar 3.23 Tampilan Daftar Santri	53
Gambar 3.24 Tampilan Daftar Penjaga	53
Gambar 3.25 Tampilan Daftar Wali	54
Gambar 3.26 Tampilan Daftar Asrama	54
Gambar 3.27 Tampilan Laporan Izin	55

DAFTAR TABEL

Tabel 2.1 Simbol <i>Use Case Diagram</i>	22
Tabel 2.2 Simbol <i>Class Diagram</i>	24
Tabel 2.3 Simbol <i>Activity Diagram</i>	25
Tabel 3.1 Tabel Pengguna	55
Tabel 3.2 Tabel Wali	56
Tabel 3.3 Tabel SantriWali.....	57
Tabel 3.4 Tabel Asrama	57
Tabel 3.5 Tabel Izin.....	57

BAB I

PENDAHULUAN

A. Latar Belakang

Pesatnya perkembangan teknologi informasi dan komunikasi dalam beberapa tahun terakhir telah membawa banyak perubahan signifikan dalam berbagai aspek kehidupan, termasuk sistem manajemen dan administrasi. Dalam konteks pendidikan dan pengelolaan asrama, penerapan teknologi informasi yang tepat tidak hanya membuat pekerjaan lebih mudah dan cepat, tetapi juga mengurangi risiko kesalahan dalam administrasi. Perguruan Diniyyah Puteri Padang Panjang, sebuah pondok pesantren modern khusus putri yang berlokasi di Kota Padang Panjang, Sumatra Barat, merasakan kebutuhan akan modernisasi sistem administrasi ini, terutama dalam hal pengelolaan izin keluar-masuk asrama.

Saat ini, proses pengelolaan izin keluar dan masuk asrama di Perguruan Diniyyah Puteri masih dilakukan secara manual menggunakan buku yang ditandatangani oleh penjaga asrama dan divalidasi oleh staf administrasi. Proses manual ini memakan banyak waktu dan tenaga, serta berpotensi menimbulkan berbagai masalah, seperti kehilangan data, duplikasi informasi, pemalsuan data, dan keterlambatan dalam pemrosesan izin. Penulis juga menemukan permasalahan dalam perizinan keluar, misalnya ketika santri ingin meminta izin keluar tetapi penjaga asrama tidak berada di tempat, sehingga santri harus meminta izin kepada kepala asrama. Akibatnya, informasi ini tidak sampai kepada penjaga asrama, menyebabkan penjaga asrama tersebut tidak mengetahui bahwa santri tersebut telah diizinkan keluar. Kondisi ini jelas membuat pekerjaan menjadi lebih rumit.

Untuk mengatasi masalah ini, diperlukan solusi yang mampu membuat pekerjaan lebih mudah, cepat dan tepat dalam pengelolaan perizinan santri. Salah satu solusinya adalah pengembangan sistem informasi berbasis web dengan memanfaatkan teknologi terbaru. Berbagai teknologi modern dapat digunakan dalam pengembangan sistem ini, termasuk *framework* seperti

Spring Boot, Laravel, Node.js, React, Angular, dan lain-lain, serta *database* seperti PostgreSQL, MySQL, MongoDB, dan lainnya. Dalam tugas akhir ini, penulis memilih untuk menggunakan *framework* Spring Boot dan *database* PostgreSQL.

Spring Boot adalah *framework* yang populer di kalangan perusahaan untuk pengembangan aplikasi berbasis Java, karena kemudahan yang ditawarkannya dalam membangun aplikasi web secara cepat dan tepat. Fitur-fitur *built-in* yang dimilikinya memudahkan pengembangan *RESTful API*, pengelolaan dependensi yang sederhana, serta dukungan yang kuat untuk aspek keamanan dan kemampuan untuk berkembang lebih jauh. (Broadcom, 2024)

Di sisi lain, PostgreSQL merupakan sistem manajemen basis data relasional *open-source* yang dikenal karena kestabilan, kemampuan untuk berkembang lebih jauh, dan kemampuannya dalam menangani berbagai jenis data dengan baik (The PostgreSQL Global Development Group, 2024). Kombinasi penggunaan Spring Boot dan PostgreSQL diharapkan dapat menghasilkan sistem informasi perizinan santri yang andal serta memberikan performa yang baik.

Dengan memanfaatkan kedua teknologi ini, sistem informasi perizinan keluar-masuk santri yang dikembangkan diharapkan tidak hanya memenuhi kebutuhan administrasi asrama, tetapi juga memberikan kemudahan akses, pemantauan, dan pengelolaan yang lebih baik, serta menjadi kontribusi nyata dalam penerapan teknologi informasi di lingkungan pendidikan. Untuk itu, penulis mengambil judul **“PERANCANGAN SISTEM INFORMASI PERIZINAN KELUAR MASUK SANTRI PADA ASRAMA PERGURUAN DINIYAH PUTERI PADANG PANJANG DENGAN MENGGUNAKAN *FRAMEWORK* SPRING BOOT DAN POSTGRESQL”**.

B. Identifikasi Masalah

Berdasarkan latar belakang di atas, dapat dirumuskani bahwa permasalahan yang dipilih sebagai pokok bahasan penelitian ini adalah:

1. Pengelolaan izin keluar dan masuk asrama di Perguruan Diniyyah Puteri Padang Panjang saat ini masih dilakukan secara manual dengan menggunakan buku, sehingga membutuhkan banyak waktu dan tenaga.
2. Penggunaan sistem manual mudah menyebabkan berbagai kesalahan, seperti kehilangan data, duplikasi informasi, atau bahkan pemalsuan data. Hal ini bisa membuat catatan perizinan santri menjadi tidak tepat dan sulit diperiksa.
3. Sistem manual membuat pemantauan dan pengelolaan proses perizinan menjadi sulit, baik untuk staf administrasi maupun pihak lain yang terlibat. Hal ini dapat menyebabkan keterlambatan dalam pengambilan keputusan mengenai izin santri.

C. Batasan Masalah

Penelitian ini difokuskan pada pengembangan sistem informasi untuk membantu pengelolaan perizinan keluar-masuk santri di Perguruan Diniyyah Puteri Padang Panjang. Agar penelitian lebih terarah, batasannya ditentukan sebagai berikut:

1. Sistem ini dibuat untuk menggantikan metode pencatatan manual berbasis buku dengan sistem berbasis web. Dengan cara ini, pengelolaan izin diharapkan menjadi lebih mudah dan cepat.
2. Sistem dirancang untuk mengurangi masalah seperti kehilangan data, duplikasi informasi, atau risiko adanya data yang tidak valid, yang kerap muncul dalam proses manual.
3. Dengan adanya sistem ini, petugas asrama dan staf administrasi akan lebih mudah memantau status izin santri, serta dapat memproses pengajuan izin dengan lebih cepat, baik itu menyetujui maupun menolaknya.

D. Rumusan Masalah

Berdasarkan uraian latar belakang masalah di atas, rumusan masalah yang melatarbelakangi tugas akhir ini adalah: Bagaimana merancang dan membangun sistem informasi berbasis web untuk mengelola izin keluar dan masuk santri di asrama Perguruan Diniyyah Puteri Padang Panjang dengan menggunakan *framework* Spring Boot dan PostgreSQL, sehingga sistem ini dapat mempermudah dan mempercepat proses perizinan, mengurangi kesalahan administrasi, serta mudah diakses, dipantau, dan dikelola oleh staf administrasi asrama, penjaga asrama, dan santri?

E. Tujuan Penelitian

Penelitian ini bertujuan untuk merancang dan membangun sistem informasi berbasis web yang dapat mengelola izin keluar dan masuk santri di asrama Perguruan Diniyyah Puteri Padang Panjang. Dengan menggunakan *framework* Spring Boot dan PostgreSQL, sistem ini diharapkan dapat:

1. Mempermudah dan mempercepat proses perizinan santri.
2. Mengurangi kesalahan administrasi yang sering terjadi dalam proses manual.
3. Menyediakan akses yang mudah dan cepat bagi staf administrasi, penjaga asrama, dan santri.
4. Memungkinkan pemantauan yang lebih mudah dan cepat terhadap proses perizinan.

F. Manfaat Penelitian

Berdasarkan tujuan penelitian yang hendak dicapai, maka penelitian ini memiliki manfaat dalam sistem informasi perizinan. Berikut adalah beberapa manfaat dari penelitian tersebut:

1. Manfaat bagi Perguruan Diniyyah Puteri Padang Panjang mencakup optimisasi pengelolaan asrama dan izin santri dengan sistem yang lebih modern dan terstruktur, yang memudahkan pengambilan keputusan berbasis data yang lebih tepat. Selain itu, sistem ini juga mengurangi waktu dan tenaga yang dibutuhkan untuk administrasi manual.

2. Manfaat bagi pengguna mencakup kemudahan akses bagi santri, penjaga asrama, dan staf administrasi dalam mengelola izin keluar-masuk melalui sistem yang mudah digunakan. Staf administrasi dapat mengurangi beban kerja mereka dengan beralih dari proses manual ke sistem otomatis. Sistem ini juga membuat pengelolaan izin lebih dapat dipantau dengan baik, sehingga santri dan orang tua mendapatkan pelayanan yang lebih baik.
3. Sebagai salah satu syarat untuk mendapatkan gelar Ahli Madya Program Diploma III (D.III) Manajemen Informatika di UIN Mahmud Yunus Batusangkar.

G. Metode Pengembangan Sistem

Untuk pengembangan Sistem Informasi Perizinan Keluar Masuk Santri pada Asrama Perguruan Diniyyah Puteri Padang Panjang, penelitian ini menggunakan metodologi *Systems Development Life Cycle* (SDLC) dengan model *Waterfall*.

Menurut Muhammad Ridwan, Iskandar Fitri, dan Benrahman (2021), SDLC adalah sebuah metodologi yang menggambarkan siklus hidup dalam pengembangan perangkat lunak. Metodologi ini berfokus pada proses pembuatan dan perbaikan sistem, khususnya sistem komputer atau sistem informasi. Dalam SDLC, terdapat berbagai model pengembangan seperti *prototyping*, *Rapid Application Development (RAD)*, *agile*, *waterfall*, *V-model*, dan beberapa model lainnya. Masing-masing model ini menawarkan pendekatan yang berbeda dalam merancang dan mengembangkan sistem.

Metodologi SDLC memberikan kerangka kerja yang jelas dalam setiap langkah pengembangan sistem, mulai dari tahap perencanaan hingga pemeliharaan. Dengan mengikuti alur ini, diharapkan sistem yang dihasilkan tidak hanya memenuhi kebutuhan pengguna, tetapi juga dapat diandalkan untuk jangka panjang.

Model *Waterfall*, yang juga dikenal sebagai model air terjun, merupakan salah satu pendekatan klasik dalam pengembangan perangkat

lunak. Model ini menekankan pendekatan berurutan, di mana setiap tahapan pengembangan harus diselesaikan sepenuhnya sebelum berlanjut ke tahap berikutnya. Tahapan dalam model *Waterfall* adalah sebagai berikut:

1. Perencanaan (*Planning*): Pada tahap ini, tujuan utama penelitian dan ruang lingkup sistem ditetapkan. Kebutuhan sistem diidentifikasi, dan strategi pengembangan dirancang serta merencanakan sumber daya yang diperlukan.
2. Analisis (*Analysis*): Dalam tahap ini, kebutuhan pengguna dan spesifikasi sistem dipahami secara mendalam. Studi kelayakan dilakukan untuk memastikan bahwa sistem yang diusulkan dapat dibangun sesuai dengan kebutuhan.
3. Desain (*Design*): Setelah analisis kebutuhan, desain sistem dilakukan. Tahap ini mencakup desain arsitektur sistem, desain *database*, antarmuka pengguna, dan komponen lain yang diperlukan untuk implementasi.
4. Implementasi (*Implementation*): Pada tahap implementasi, kode program ditulis berdasarkan desain yang telah dibuat. Sistem informasi kemudian dibangun dan dipersiapkan untuk digunakan.

H. Sistematika Penulisan

Sistematika penulisan adalah proses pembuatan laporan yang menyajikan rangkuman informasi dalam tugas akhir. Sistematika penulisan terdiri dari pendahuluan, landasan teori, analisis dan hasil, serta kesimpulan. Empat bab yang menyusun sistem penulisan laporan akhir disusun sebagai berikut:

1. BAB I PENDAHULUAN

Bab ini mencakup:

- a. Latar Belakang Masalah: Penjelasan mengenai latar belakang yang melatarbelakangi penelitian atau proyek yang dilakukan.
- b. Identifikasi Masalah: Pengidentifikasian masalah-masalah yang dihadapi yang relevan dengan penelitian atau proyek.

- c. Batasan Masalah: Penjelasan mengenai batasan-batasan yang diterapkan dalam penelitian atau proyek untuk memfokuskan ruang lingkup.
- d. Rumusan Masalah: Rumusan masalah yang akan dijawab oleh penelitian atau proyek.
- e. Tujuan Penelitian: Penjelasan mengenai tujuan dari penelitian atau proyek yang dilakukan.
- f. Manfaat Penelitian: Uraian mengenai manfaat yang diharapkan dari hasil penelitian atau proyek, baik untuk pengembangan ilmu pengetahuan, praktik lapangan, maupun manfaat praktis lainnya.
- g. Metodologi Penelitian: Uraian tentang metode penelitian atau pendekatan yang digunakan dalam penelitian atau proyek.
- h. Sistematika Penulisan: Penjelasan mengenai sistematika penulisan yang digunakan dalam laporan.

2. BAB II LANDASAN TEORI

Bab ini berisi penjelasan mengenai konsep dan teori yang digunakan dalam perancangan sistem informasi perizinan menggunakan *framework* Spring Boot dan PostgreSQL. Bagian ini mencakup:

- a. Konsep Dasar: Penjelasan mengenai konsep dasar yang mendasari penelitian.
- b. Teori Terkait: Uraian tentang teori-teori yang relevan dengan penelitian.
- c. Teknologi yang Digunakan: Penjelasan mengenai teknologi-teknologi yang digunakan, seperti Spring Boot dan PostgreSQL.

3. BAB III ANALISIS DAN HASIL

Bab ini berisi uraian mengenai hasil analisis kebutuhan dan perancangan sistem informasi perizinan menggunakan *framework* Spring Boot dan PostgreSQL. Bagian ini mencakup:

- a. Analisis Kebutuhan: Uraian mengenai kebutuhan sistem yang diidentifikasi selama penelitian atau proyek.

- b. Perancangan Sistem: Penjelasan mengenai perancangan sistem yang dilakukan berdasarkan analisis kebutuhan.
- c. Implementasi: Uraian tentang implementasi sistem informasi perizinan yang dilakukan.
- d. Hasil dan Pembahasan: Penjelasan mengenai hasil dari implementasi sistem serta pembahasan mengenai temuan-temuan yang diperoleh.

4. BAB IV PENUTUP

Bab ini berisi tentang kesimpulan yang dicapai selama penyusunan laporan akhir proyek dan saran yang akan digunakan dalam pengembangan sistem selanjutnya. Bagian ini mencakup:

- a. Kesimpulan: Ringkasan mengenai temuan utama dan hasil yang diperoleh dari penelitian atau proyek.
- b. Saran: Rekomendasi untuk pengembangan sistem selanjutnya atau penelitian lebih lanjut yang bisa dilakukan.

BAB II

LANDASAN TEORI

A. Gambaran Umum Perguruan Diniyyah Puteri Padang Panjang

1. Sejarah Instansi

Berdasarkan catatan sejarah Perguruan Diniyyah Puteri Padang Panjang (2024), lembaga ini berdiri pada 1 November 1923 atas inisiatif Rahmah El Yunusiyah, seorang tokoh perempuan inspiratif asal Sumatera Barat. Pada masa itu, pendidikan formal bagi perempuan masih sangat terbatas. Terinspirasi oleh kakaknya, Zainuddin Labay El Yunusy, yang mendirikan Diniyyah School pada tahun 1915, Rahmah menciptakan sebuah lembaga pendidikan khusus perempuan yang memadukan nilai-nilai Islam dengan pendekatan modern.

Kegiatan belajar mengajar pada awalnya diadakan secara sederhana dengan jumlah santri yang sedikit. Namun, semangat Rahmah untuk memberdayakan perempuan melalui pendidikan menjadikan lembaga ini berkembang pesat. Kini, Perguruan Diniyyah Puteri telah menjadi salah satu institusi pendidikan Islam terkemuka di Indonesia, melayani jenjang pendidikan mulai dari PAUD hingga perguruan tinggi.

2. Visi dan Misi

Masih merujuk pada Perguruan Diniyyah Puteri Padang Panjang (2024), lembaga ini bercita-cita menjadi pusat pendidikan Islam modern yang mengintegrasikan ilmu agama dan ilmu pengetahuan. Dengan visi ini, perguruan berkomitmen mencetak generasi muslimah yang tidak hanya berkarakter mulia, tetapi juga memiliki wawasan global serta kemampuan untuk memimpin.

Untuk mewujudkan visi tersebut, Perguruan Diniyyah Puteri memiliki beberapa misi utama, yaitu:

- a) Membangun dan mengembangkan berbagai pusat keunggulan pengetahuan bersumberkan kepada Al Qur'an dan Hadits

- b) Menyiapkan Sumber Daya Manusia yang berkualitas global dan berakhlak sesuai syariah
- c) Mengembangkan kegiatan usaha secara profesional untuk mencapai kemandirian berkarya

3. Profil Instansi

Perguruan Diniyyah Puteri berlokasi di Jl. Abdul Hamid Hakim No. 29, Silaing Bawah, Kota Padang Panjang, Sumatera Barat. Terletak di daerah pegunungan, lingkungan perguruan ini memiliki suasana yang sejuk dan asri, memberikan suasana belajar yang mendukung bagi para santri. Lokasi strategis ini menjadi salah satu alasan banyak orang tua memilih Perguruan Diniyyah Puteri untuk anak-anak mereka (Perguruan Diniyyah Puteri Padang Panjang, 2024).

B. Konsep Dasar Sistem Informasi

1. Sistem

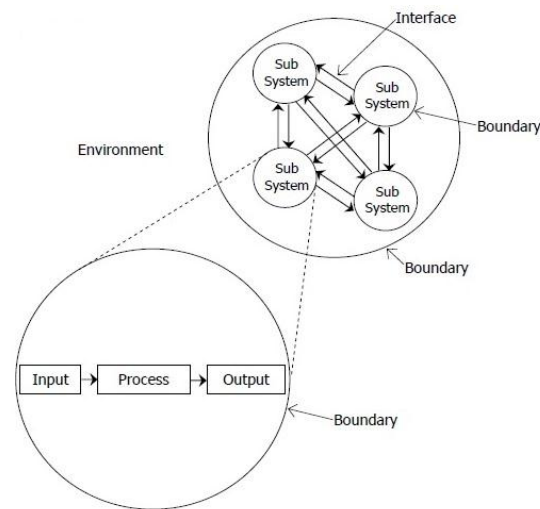
a. Pengertian Sistem

Sistem adalah konsep yang kompleks, tetapi dapat dijelaskan dengan sederhana. Menurut KBBI (2023), sistem adalah kumpulan unsur yang saling terkait dan bekerja sama untuk mencapai tujuan tertentu. Dalam kata lain, sistem adalah perangkat yang terdiri dari beberapa komponen yang memiliki hubungan dan bekerja sama untuk mencapai hasil yang diinginkan.

Para ahli juga memiliki definisi yang serupa tentang sistem. Satzinger, Jackson, dan Bard mendefinisikan sistem sebagai kumpulan komponen yang memiliki hubungan dan bekerja sama untuk mencapai tujuan yang ditentukan. Sementara itu, Murdick mendefinisikan sistem sebagai perangkat elemen atau pengelolaan yang terdiri dari kegiatan dan prosedur yang sama dalam menjalankan data dengan waktu yang ditentukan, sehingga menghasilkan informasi, energi, atau barang. (Prehanto, 2020)

Dengan demikian, dapat disimpulkan bahwa sistem adalah kumpulan komponen yang saling terkait dan bekerja sama untuk mencapai tujuan tertentu, dengan struktur dan organisasi yang terdefinisi dengan jelas. Sistem dapat membantu kita mencapai hasil yang diinginkan dengan lebih mudah dan lancar.

b. Karakteristik Sistem



Gambar 2.1 Karakteristik Sistem

Karakteristik yang harus dimiliki dalam sebuah sistem, sesuai dengan gambar di atas, adalah sebagai berikut:

- 1) Sistem terdiri dari beberapa komponen yang saling berinteraksi dan berfungsi sebagai satu kesatuan yang saling bekerja sama.
- 2) Lingkungan luar sistem (*Environment*)

Ini adalah faktor-faktor dari luar yang memengaruhi operasi sistem, baik yang bersifat menguntungkan maupun merugikan. Meskipun demikian, lingkungan ini tetap perlu dijaga dan dikelola dengan baik.

- 3) Batasan sistem (*Boundary*)

Batasan sistem merupakan garis yang memisahkan sistem dari yang lainnya atau dari lingkungan luar, sesuai dengan ruang lingkup yang ditetapkan.

- 4) Penghubung sistem (*Interface*)

Penghubung sistem adalah mekanisme yang menjembatani satu subsistem dengan subsistem lainnya. Fungsinya adalah memastikan aliran sumber daya antar-subsistem berjalan lancar. Dengan adanya penghubung ini, *output* dari satu subsistem dapat langsung digunakan sebagai *input* untuk subsistem lain, sehingga menciptakan koordinasi yang lebih mudah dan metode yang baik dalam keseluruhan sistem.

5) Masukkan sistem (*Input*)

Input adalah segala sumber daya yang dimasukkan ke dalam sistem untuk mendukung proses operasionalnya. *Input* ini terdiri dari dua jenis yang mana kedua jenis ini memiliki peran penting dalam menjaga keberlanjutan dan hasil yang diinginkan dari sistem:

- a) Perawatan (*Maintenance Input*): *Input* yang berfungsi untuk menjaga agar sistem tetap berjalan dengan baik. Contohnya adalah perangkat lunak (*software*) yang diperbarui secara berkala untuk memastikan sistem tetap berfungsi optimal.
- b) Sinyal (*Signal Input*): *Input* yang diproses oleh sistem untuk menghasilkan *output*. Sebagai contoh, data sinyal yang digunakan oleh proyektor untuk menampilkan gambar.

6) Keluaran sistem (*Output*)

Output adalah hasil berupa energi atau informasi yang dihasilkan oleh sistem setelah memproses *input*. *Output* dapat berupa:

- a) Energi yang Dibuang: Hasil yang tidak dimanfaatkan, misalnya suhu panas yang dihasilkan oleh komputer saat beroperasi.
- b) Energi yang Digunakan: Hasil yang diperlukan dan dimanfaatkan, seperti informasi yang ditampilkan oleh komputer.

7) Pengolah sistem

Pengolah sistem adalah bagian dari sistem yang bertugas mengubah *input* menjadi *output*. Proses ini merupakan inti dari operasional sistem untuk menghasilkan hasil yang diinginkan. Pengolah sistem memastikan setiap *input* diproses secara efisien untuk menghasilkan *output* yang sesuai dengan tujuan sistem.

8) Sasaran sistem

Sasaran sistem adalah tujuan atau hasil akhir yang ingin dicapai oleh sebuah sistem. Sasaran ini menjadi pedoman utama dalam merancang dan menjalankan sistem untuk memastikan semua proses berjalan sesuai dengan yang diharapkan.

c. **Klasifikasi Sistem**

Sistem dapat diklasifikasikan berdasarkan sudut pandang tertentu, seperti berikut:

- 1) Sistem Abstrak (*Abstract System*): Sistem ini berupa gagasan atau ide yang tidak terlihat secara fisik. Contohnya adalah sistem teologi, yaitu konsep hubungan antara manusia dengan Tuhan yang berbentuk pemikiran atau keyakinan.
- 2) Sistem Fisik (*Physical System*): Sistem Fisik adalah sistem yang memiliki bentuk fisik dan bisa dilihat secara langsung. Sistem ini dirancang untuk memenuhi kebutuhan tertentu. Contoh dari sistem fisik adalah komputer, mesin, dan perangkat lunak.
- 3) Sistem Tertentu (*Deterministic System*): Sistem ini berjalan otomatis dengan hasil yang dapat diprediksi karena mengikuti aturan atau program tertentu. Contohnya adalah sistem prediksi cuaca

2. **Informasi**

a. **Pengertian Informasi**

Informasi adalah sesuatu yang sangat penting dalam kehidupan sehari-hari kita. Menurut KBBI (2023), informasi adalah pemberitahuan, kabar, atau berita tentang sesuatu yang dapat

membantu kita memahami dan mengerti tentang hal-hal yang terjadi di sekitar kita.

Namun, informasi tidak hanya sekedar pemberitahuan atau kabar. Menurut McLeod, seperti yang dikutip oleh Yakub dalam buku *Pengertian Sistem Informasi*, informasi adalah hasil dari pengelolaan data yang membuatnya menjadi lebih berguna dan berarti bagi penerimanya (Prehanto, 2020). Dengan kata lain, informasi adalah hasil dari proses pengolahan data yang membuatnya menjadi lebih mudah dipahami dan digunakan.

Sementara itu, Surabri mendefinisikan informasi sebagai hasil dari pengelolaan data yang diinterpretasikan dan diklasifikasi, yang kemudian digunakan dalam proses pengambilan keputusan (Prehanto, 2020). Ini berarti bahwa informasi tidak hanya sekedar data yang dikumpulkan, tetapi juga harus diolah dan diinterpretasikan agar dapat digunakan dengan efektif.

Dengan demikian, dapat disimpulkan bahwa informasi adalah hasil dari pengelolaan data yang membuatnya menjadi lebih berguna dan berarti bagi penerimanya, dan dapat digunakan dalam proses pengambilan keputusan yang lebih baik.

b. Jenis-jenis Informasi

Berikut adalah beberapa jenis informasi beserta penjelasannya:

- 1) *Absolute Information*: Informasi ini bersifat pasti dan tidak memerlukan penjelasan tambahan. Informasi ini dianggap sebagai induk dari semua informasi yang ada.
- 2) *Substitutional Information*: Informasi yang menggunakan konsep yang dapat diterapkan pada berbagai jenis informasi lain. *Substitutional information* juga sering disebut sebagai komunikasi.
- 3) *Philosophic Information*: Jenis informasi ini menghubungkan pengetahuan dengan kebijaksanaan. Filosofi di balik informasi ini bertujuan untuk memberikan wawasan yang lebih mendalam.

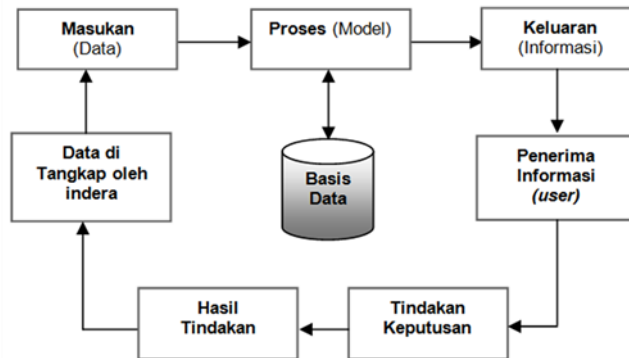
- 4) *Subjective Information*: Informasi yang dipengaruhi oleh perasaan atau sudut pandang individu yang menyampaikan. Informasi ini sangat bergantung pada penyaji atau pengirimnya.
- 5) *Objective Information*: Informasi yang berfokus pada fakta logis dan data yang terukur tanpa dipengaruhi oleh opini atau emosi.
- 6) *Cultural Information*: Informasi yang berkaitan dengan aspek budaya, menyoroti nilai, tradisi, dan kebiasaan dalam suatu masyarakat.

c. Siklus Informasi

Siklus informasi dimulai dari data yang berfungsi sebagai *input*. Data ini kemudian diproses menggunakan model tertentu untuk menghasilkan *output* yang berguna. Proses pengolahan data ini bertujuan agar informasi yang dihasilkan bisa bermanfaat bagi penerima, baik dalam pengambilan keputusan, pelaksanaan kegiatan, maupun evaluasi.

Data yang belum diolah biasanya disimpan dalam bentuk basis data. Data ini dapat diakses kembali untuk diproses menjadi informasi yang diperlukan.

Proses ini membentuk sebuah siklus, di mana data yang diproses menjadi informasi, yang kemudian digunakan oleh penerima untuk membuat keputusan dan mengambil tindakan, yang pada gilirannya akan menghasilkan data baru untuk diproses. Siklus ini dikenal sebagai Siklus Informasi (*Information Cycle*).



Gambar 2.2 Siklus Informasi

3. Sistem Informasi

a. Pengertian Sistem Informasi

Sistem informasi adalah jantung dari sebuah organisasi, karena itu membantu pembuat keputusan yang tepat dan mengendalikan operasional dengan lebih baik. Secara sederhana, sistem informasi adalah kumpulan prosedur yang diorganisasikan untuk menyediakan informasi yang dibutuhkan oleh organisasi untuk membuat keputusan yang tepat.

Dalam bahasa yang lebih luas, sistem informasi terdiri dari kumpulan orang yang bekerja sama dan secara terstruktur untuk mencapai tujuan-tujuan tertentu. Seperti yang dijelaskan oleh Dwipa, Oktarina, Teisnajaya, dan Pangestu (2023), sistem informasi adalah kumpulan orang yang saling bekerja sama untuk memenuhi tujuan-tujuan tertentu.

Dengan demikian, dapat disimpulkan bahwa sistem informasi adalah kumpulan komponen teknologi informasi yang mengelola data sehingga data yang diolah dapat dijadikan sebagai informasi yang bermanfaat dan dapat membantu untuk mencapai tujuan organisasi atau kelompok. Sistem informasi membantu kita mengubah data menjadi informasi yang dapat digunakan untuk membuat keputusan yang tepat dan mencapai tujuan yang diinginkan.

Berikut adalah beberapa komponen yang diperlukan pada sistem informasi untuk membuatnya berfungsi dengan efektif:

- 1) Perangkat Keras (*Hardware*): Komponen fisik seperti komputer, server, perangkat jaringan, dan alat lainnya yang digunakan untuk memproses dan menyimpan data dalam sistem informasi.
- 2) Perangkat Lunak (*Software*): Program komputer dan aplikasi yang dirancang untuk mengelola data dan menjalankan berbagai proses bisnis.
- 3) Data: Informasi mentah yang dihasilkan oleh individu atau organisasi, berupa angka, teks, gambar, atau bentuk informasi lainnya yang akan diolah menjadi informasi berguna.
- 4) Prosedur: Serangkaian langkah atau pedoman yang digunakan untuk mengelola data dan memastikan proses bisnis berjalan secara terstruktur.
- 5) Manusia (*Brainware*): Pengguna dan administrator sistem informasi. Mereka bertanggung jawab atas pengumpulan data, *input*, pengolahan, analisis, dan pengambilan keputusan berbasis informasi.

b. Fungsi Sistem Informasi

Sistem informasi memiliki beberapa fungsi utama yang mendukung kelancaran operasional dan pengambilan keputusan, yaitu:

1) Mengumpulkan Data

Sistem informasi bertugas mengumpulkan data mentah dari berbagai sumber, baik dari dalam (internal) maupun luar (eksternal) organisasi, sesuai dengan kebutuhan operasional atau strategis.

2) Mengolah Data

Data yang telah dikumpulkan diproses menggunakan metode tertentu untuk menghasilkan informasi yang bermakna, relevan, dan dapat langsung digunakan.

3) Menyimpan Data

Sistem informasi membantu menyimpan data dengan aman dalam basis data untuk penggunaan jangka panjang atau kebutuhan mendatang.

4) Mendistribusikan Informasi

Sistem informasi memungkinkan informasi yang dihasilkan dapat disampaikan kepada pihak-pihak yang membutuhkan, baik di tingkat operasional maupun manajerial, secara lebih mudah dan tepat waktu.

5) Mendukung Keputusan

Sistem ini memberikan informasi yang berkaitan dan tepat untuk membantu pengguna lain dalam mengambil keputusan yang lebih baik.

C. Konsep Perizinan Keluar Masuk Santri di Asrama

Sistem informasi perizinan memiliki peran yang sangat penting dalam mengatur aktivitas santri di asrama, terutama terkait dengan izin keluar masuk. Proses perizinan ini bukan hanya sekadar formalitas, tetapi merupakan cara bagi pengelola asrama untuk menjaga ketertiban dan keamanan lingkungan, serta memberikan perlindungan kepada santri.

Menurut Spelt dan ten Berge, izin dapat diartikan sebagai persetujuan yang diberikan oleh otoritas berdasarkan hukum atau peraturan yang berlaku (Pudyatmoko, 2024). Dalam konteks perizinan keluar masuk santri di asrama, izin ini sangat penting untuk mengatur aktivitas santri yang ingin pergi atau kembali. Hal ini menunjukkan bahwa santri tidak bisa sembarangan keluar atau masuk tanpa mendapatkan izin dari pengelola asrama. Dengan demikian, pengelola asrama memiliki peran yang krusial dalam mengawasi dan mengatur kegiatan santri, memastikan bahwa semua aktivitas berlangsung sesuai dengan aturan yang telah ditetapkan.

Pandangan ini sedikit berbeda dari yang disampaikan oleh Van der Pot, yang menyatakan bahwa izin adalah keputusan yang membolehkan tindakan

yang pada dasarnya tidak dilarang oleh pembuat peraturan (Pudyatmoko, 2024). Dalam konteks asrama, ini berarti santri diizinkan untuk keluar masuk asrama selama tindakan tersebut tidak melanggar aturan yang telah ditetapkan oleh lembaga pendidikan.

Di sisi lain, Prajudi Atmosudirdjo mendefinisikan izin (*vergunning*) sebagai suatu penetapan yang memberikan kelonggaran dari larangan yang ditetapkan oleh undang-undang (Pudyatmoko, 2024). Dalam hal ini, izin keluar masuk santri dapat dilihat sebagai bentuk kelonggaran yang diberikan oleh pengelola asrama, sehingga santri dapat melakukan aktivitas di luar asrama sambil tetap mematuhi aturan yang ada.

Dengan demikian, perizinan keluar masuk santri di asrama berfungsi sebagai mekanisme kontrol yang sangat penting untuk menjaga keamanan dan ketertiban di lingkungan asrama. Izin ini tidak hanya melindungi santri itu sendiri, tetapi juga menciptakan suasana yang aman dan teratur bagi semua penghuni asrama.

Adapun tujuan dari perizinan keluar masuk santri di asrama adalah sebagai berikut ini:

1. Mengarahkan Aktivitas Santri

Pengelola asrama menggunakan izin untuk mengarahkan aktivitas santri yang ingin keluar atau masuk. Misalnya, ketika seorang santri ingin pergi ke suatu tempat, mereka harus mengajukan permohonan izin dengan alasan yang jelas. Proses ini membantu pengelola untuk memantau dan mengatur kegiatan santri, sehingga tidak terjadi penyalahgunaan izin. Dengan adanya sistem ini, pengelola dapat memastikan bahwa santri tetap berada dalam batas yang aman dan sesuai dengan ketentuan yang telah ditetapkan.

2. Mencegah Bahaya

Sistem informasi perizinan juga berfungsi untuk mencegah potensi bahaya yang mungkin dihadapi santri saat berada di luar asrama. Dengan mengevaluasi alasan dan tujuan santri yang ingin keluar, pengelola dapat mengambil langkah-langkah untuk melindungi mereka dari situasi yang

berisiko. Misalnya, jika seorang santri ingin pergi ke tempat yang dianggap tidak aman, pengelola dapat menolak permohonan izin demi keselamatan santri. Ini adalah bagian dari tanggung jawab pengelola untuk menjaga keamanan dan kenyamanan seluruh penghuni asrama.

3. Melindungi Santri

Pengelola asrama juga memiliki tanggung jawab untuk melindungi santri dari berbagai risiko yang mungkin terjadi saat mereka berada di luar. Dengan menerapkan sistem informasi perizinan, pengelola dapat memastikan bahwa santri tidak terlibat dalam aktivitas yang dapat membahayakan diri mereka sendiri atau orang lain. Misalnya, izin keluar dapat diberikan dengan syarat tertentu, seperti harus kembali sebelum waktu yang ditentukan, untuk memastikan bahwa santri tetap dalam pengawasan.

4. Mengelola Sumber Daya

Dalam konteks asrama, izin keluar masuk juga berfungsi untuk mengelola sumber daya yang terbatas, seperti waktu dan fasilitas. Dengan adanya izin, pengelola dapat mengatur jadwal dan jumlah santri yang keluar pada waktu tertentu, sehingga tidak terjadi penumpukan atau kekurangan fasilitas di asrama. Hal ini penting untuk menjaga kenyamanan dan keteraturan di lingkungan asrama.

5. Seleksi Aktivitas

Sistem informasi perizinan juga berfungsi untuk menyeleksi santri yang ingin keluar. Misalnya, santri yang ingin pergi ke acara tertentu harus mendapatkan izin dari orang tua atau wali. Proses ini memastikan bahwa santri yang keluar adalah mereka yang telah mendapatkan persetujuan dan dianggap layak untuk melakukan aktivitas di luar asrama. Dengan demikian, pengelola dapat menjaga keamanan dan ketertiban di lingkungan asrama.

6. Tujuan Administratif

Selain tujuan-tujuan di atas, perizinan keluar masuk santri juga mencakup aspek administratif, seperti pengaturan waktu keluar dan

masuk. Hal ini penting untuk memastikan bahwa tidak ada santri yang keluar secara bersamaan dalam jumlah besar, yang dapat mengganggu operasional asrama. Dengan adanya sistem informasi perizinan yang baik, pengelola dapat mengatur dan mengawasi aktivitas santri dengan lebih mudah.

D. Alat Bantu Perancangan Sistem

Alat bantu perancangan sistem adalah berbagai metode dan perangkat yang digunakan untuk mendukung proses analisis, desain, dan pengembangan sistem informasi. Alat-alat ini sangat membantu untuk membuat sistem yang bekerja dengan baik dan sesuai dengan kebutuhan pengguna.. Berikut adalah beberapa alat bantu digunakan dalam perancangan sistem:

1. *Unified Modelling Language (UML)*

Unified Modeling Language (UML) adalah sebuah bahasa pemodelan yang dirancang untuk membantu dalam pengembangan dan dokumentasi sistem perangkat lunak. UML menyediakan seperangkat notasi dan diagram yang memungkinkan pengembang untuk menggambarkan berbagai aspek dari sistem secara visual. Dengan menggunakan UML, proses perancangan sistem menjadi lebih terstruktur dan mudah dipahami, baik oleh pengembang maupun oleh pemangku kepentingan lainnya.

Menurut Rosa dan Shalahuddin, UML merupakan sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek (Marlindawati, 2022). Dengan memanfaatkan teknik ini, pengembang dapat menciptakan representasi visual dari sistem yang sedang dikembangkan, sehingga memudahkan dalam menggambarkan berbagai komponen dan interaksi dalam sistem secara jelas dan terstruktur.

UML terdiri dari dua kategori utama diagram: diagram struktural dan diagram perilaku. Diagram struktural, seperti diagram kelas dan diagram komponen, berfungsi untuk menggambarkan elemen-elemen


statis dalam sistem, termasuk hubungan antar objek dan struktur data. Di sisi lain, diagram perilaku, seperti diagram urutan dan diagram aktivitas, digunakan untuk menggambarkan interaksi antar komponen dan alur kerja dalam sistem. Berikut beberapa contoh dari masing-masing diagram tersebut:

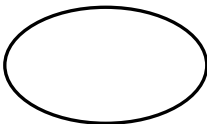



a. Use Case Diagram

Use Case Diagram adalah salah satu jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan interaksi antara pengguna (aktor) dan sistem yang sedang dikembangkan. Diagram ini berfungsi untuk menunjukkan fungsionalitas sistem dari sudut pandang pengguna, serta menggambarkan bagaimana aktor berinteraksi dengan berbagai *use case* yang ada dalam sistem.

Menurut Rosa dan Shalahuddin, *Use Case Diagram* merupakan pemodelan untuk kelakuan sistem yang dibuat, di mana *use case* mendeskripsikan interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat (Marlindawati, 2022). Dengan kata lain, *Use Case Diagram* membantu dalam mengidentifikasi dan mendokumentasikan kebutuhan fungsional sistem, sehingga pengembang dapat memahami apa yang diharapkan oleh pengguna. Dalam *Use Case Diagram*, terdapat beberapa elemen penting, yaitu:

Tabel 2.1 Simbol *Use Case Diagram*

NO	SIMBOL	NAMA	KETERANGAN
1		Aktor	Mewakili peran orang, sistem lain, atau alat yang berinteraksi dengan sistem. Aktor dapat berupa pengguna akhir, administrator, atau sistem eksternal yang berkomunikasi dengan


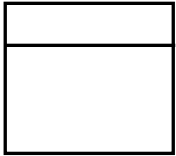



			sistem yang sedang dikembangkan.
2		<i>Use Case</i>	Representasi dari fungsionalitas atau layanan yang disediakan oleh sistem. Setiap <i>use case</i> menggambarkan satu skenario interaksi antara aktor dan sistem.
3		Relasi Asosiasi	Hubungan dasar yang menunjukkan interaksi langsung antara aktor dan use case.
4	<p><<extend>></p> 	<i>Extend</i>	Relasi <i>extend</i> digunakan untuk menunjukkan bahwa fungsionalitas tambahan (<i>use case</i> yang di- <i>extend</i>) dapat ditambahkan ke <i>use case</i> yang lain (<i>use case</i> yang diperluas) dalam kondisi tertentu.
5	<p><<include>></p> 	<i>Include</i>	Relasi <i>include</i> digunakan untuk menunjukkan bahwa satu <i>use case</i> (<i>use case</i> yang di- <i>include</i>) selalu dipanggil atau digunakan oleh <i>use case</i> lain (<i>use case</i> yang memanggil).


b. *Class Diagram*

Menurut Rosa dan Shalahuddin, *class diagram* berfungsi untuk menunjukkan struktur dari suatu sistem serta mendefinisikan kelas-kelas yang akan dibuat dalam proses pengembangan sistem tersebut (Marlindawati, 2022). Dengan kata lain, diagram ini memberikan

gambaran jelas tentang bagaimana berbagai komponen dalam sistem saling terhubung dan berinteraksi melalui kelas-kelas yang telah ditentukan. Berikut simbol-simbol yang ada pada *Class Diagram*:

Tabel 2.2 Simbol *Class Diagram*



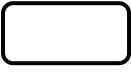

NO	SIMBOL	NAMA	KETERANGAN
1		<i>Generalization</i>	Proses mengelompokkan beberapa <i>class</i> yang memiliki karakteristik atau atribut yang sama ke dalam satu <i>class</i> yang lebih umum.
2		<i>Class</i>	Elemen dasar dalam <i>Class Diagram</i> yang berfungsi sebagai cetak biru untuk objek-objek yang akan dibuat.
3		<i>Collaboration</i>	<i>Collaboration</i> dalam <i>Class Diagram</i> menggambarkan bagaimana beberapa kelas atau objek bekerja sama untuk mencapai tujuan tertentu. Ini menunjukkan interaksi antara objek-objek dalam sistem.
4		<i>Realization</i>	Proses di mana sebuah <i>class</i> konkret mengimplementasikan antarmuka atau <i>class</i> abstrak.
5		<i>Dependency</i>	Hubungan di mana satu <i>class</i> bergantung pada <i>class</i> lain untuk berfungsi dengan baik.

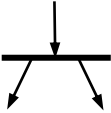
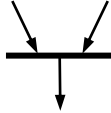
6		<i>Association</i>	Hubungan antara dua atau lebih <i>class</i> yang menunjukkan bagaimana mereka saling terhubung. Hubungan ini bisa berupa satu-ke-satu, satu-ke-banyak, atau banyak-ke-banyak.
---	---	--------------------	---

c. Activity Diagram

Menurut Rosa dan Shalahuddin, diagram aktivitas, atau yang dikenal sebagai *activity diagram*, berfungsi untuk menggambarkan alur kerja atau aktivitas dalam suatu sistem, proses bisnis, atau menu yang terdapat dalam perangkat lunak (Marlindawati, 2022). Penting untuk dicatat bahwa diagram ini fokus pada aktivitas yang dilakukan oleh sistem itu sendiri, bukan pada tindakan yang dilakukan oleh aktor. Dengan kata lain, diagram aktivitas menyoroti bagaimana sistem menjalankan fungsinya dan mengelola proses yang ada. Berikut simbol-simbol yang ada pada *Activity Diagram*:

Tabel 2.3 Simbol *Activity Diagram*

NO	SIMBOL	NAMA	KETERANGAN
1		Status Awal	Titik awal dari suatu proses atau aktivitas.
2		Status Akhir	Simbol ini menandakan bahwa suatu proses atau aktivitas telah selesai.
3		<i>Activities</i> (Aktifitas)	Tindakan atau langkah yang dilakukan dalam suatu proses.
4		<i>Decision</i> (Keputusan)	Simbol ini digunakan untuk menunjukkan titik di mana sebuah

			keputusan harus diambil.
5		<i>Fork Node</i> (Percabangan)	Satu alur aktivitas dapat dibagi menjadi beberapa jalur yang berjalan secara bersamaan.
6		<i>Join Node</i> (Penggabungan)	Simbol ini menandakan bahwa beberapa jalur aktivitas yang sebelumnya terpisah akan bergabung kembali menjadi satu alur.

E. Perangkat Lunak Pembangunan Sistem

1. Java

a. Pengertian Java

Java adalah teknologi pemrograman yang pertama kali diperkenalkan oleh Sun Microsystems pada pertengahan tahun 1990-an. Secara definisi, Java adalah kumpulan teknologi yang dirancang untuk membuat dan menjalankan perangkat lunak pada komputer standalone maupun dalam lingkungan jaringan. Salah satu aspek utama Java adalah sifatnya yang portable. Program Java dapat dijalankan di berbagai platform sistem operasi selama sistem tersebut mendukung *Java Virtual Machine* (JVM). JVM bertugas membaca bytecode (file .class) hasil kompilasi dari kode sumber Java, yang merupakan representasi langsung dari bahasa mesin. Hal ini menjadikan Java dikenal sebagai bahasa pemrograman lintas platform yang fleksibel dan efisien.

(Warno, 2020)

b. Karakteristik Utama Java

Menurut Warno (2020), beberapa karakteristik utama Java yang membuatnya unggul dibandingkan bahasa pemrograman lain adalah sebagai berikut:

- 1) *Platform Independence*: Program yang ditulis dalam Java dapat dijalankan di berbagai sistem operasi tanpa memerlukan perubahan kode. Misalnya, aplikasi yang dibuat di Windows dapat dijalankan di Linux atau macOS tanpa modifikasi.
- 2) Berorientasi Objek (*Object-Oriented Programming*): Java sepenuhnya mendukung paradigma pemrograman berorientasi objek. Paradigma ini membantu pengembang untuk memodelkan dunia nyata dalam bentuk program melalui konsep seperti kelas, objek, pewarisan, dan polimorfisme.
- 3) Kemudahan Manajemen Memori: Java memiliki sistem pengumpulan sampah otomatis (*automatic garbage collection*) yang membersihkan memori dari objek yang tidak lagi digunakan.
- 4) Keamanan Tinggi: Java dirancang dengan fokus pada keamanan. JVM dan pustaka Java menyediakan mekanisme perlindungan untuk mencegah akses yang tidak sah ke sistem.
- 5) Dukungan *Library* yang Lengkap: Java memiliki beragam *library* bawaan yang mempermudah pengembangan aplikasi. Selain itu, komunitas Java yang besar terus menyediakan pustaka tambahan untuk kebutuhan tertentu.
- 6) Sintaks yang Familiar: Sintaks Java sangat mirip dengan C++, sehingga mempermudah programmer C++ untuk beralih ke Java.

2. Spring Boot

Spring Boot adalah framework yang dirancang untuk menyederhanakan proses pengembangan aplikasi berbasis Java, khususnya dalam pembuatan REST API. Dengan Spring Boot,

pengembang dapat dengan mudah membangun aplikasi yang responsif dan efisien tanpa harus terjebak dalam konfigurasi yang rumit. Framework ini menyediakan berbagai fitur yang mendukung pengembangan aplikasi, seperti pengelolaan dependensi yang otomatis dan pengaturan yang minimal, sehingga memungkinkan pengembang untuk fokus pada logika bisnis aplikasi. (Khaqiqi & Harani, 2023)

Salah satu keunggulan utama dari Spring Boot adalah kemampuannya untuk mendukung pengembangan aplikasi berbasis RESTful. REST (Representational State Transfer) adalah arsitektur yang memungkinkan komunikasi antara klien dan server dengan cara yang sederhana dan terstruktur. Dengan menggunakan Spring Boot, pengembang dapat dengan cepat membuat endpoint yang dapat diakses oleh berbagai jenis klien, termasuk aplikasi web dan mobile.

Dalam konteks penerapan metode gamifikasi, Spring Boot memungkinkan integrasi elemen-elemen permainan ke dalam aplikasi, yang dapat meningkatkan keterlibatan pengguna. Dengan memanfaatkan fitur-fitur yang ada dalam Spring Boot, pengembang dapat menciptakan pengalaman pengguna yang lebih interaktif dan menarik, sehingga mendorong pengguna untuk berpartisipasi lebih aktif dalam aplikasi yang mereka gunakan.

3. ReactJS

a. Pengertian ReactJS

ReactJS adalah sebuah library JavaScript yang sangat terkenal untuk menciptakan antarmuka pengguna. Diciptakan oleh Facebook, library ini digunakan untuk mengembangkan aplikasi web interaktif yang kompleks. Dengan ReactJS, para pengembang dapat membuat komponen UI yang dapat digunakan kembali dan mengelola state dari komponen tersebut. Konsep yang diusung oleh ReactJS memungkinkan pengembang untuk membangun komponen UI yang efisien dan mengelola Virtual DOM untuk rendering elemen UI yang lebih cepat.

ReactJS sering dipadukan dengan library JavaScript lainnya untuk membangun aplikasi web yang komprehensif. (Kusnawi et al., 2024)

b. Fundamental ReactJS

Menurut Kusnawi, Romadhon, dan Dhuhita, Windha (2024), ada beberapa elemen dasar yang perlu dipahami untuk menguasai cara kerja aplikasi dengan ReactJS. Berikut adalah beberapa konsep utama dalam ReactJS:

1) Komponen

ReactJS mengadopsi arsitektur berbasis komponen, di mana antarmuka pengguna dibangun dengan menciptakan komponen yang dapat digunakan kembali. Terdapat dua jenis komponen utama dalam ReactJS:

- a) *Class Component*: Ini adalah class JavaScript yang mewarisi dari class `React.Component`. Class ini memungkinkan pengembang untuk memanfaatkan berbagai method yang telah disediakan, seperti `render()` yang menghasilkan kode JSX. Class component berfungsi secara otomatis.
- b) *Functional Component*: Ini adalah fungsi JavaScript yang mengembalikan JSX untuk menentukan struktur dan konten komponen. Dengan menggunakan functional component, pengembang dapat membuat komponen UI yang lebih sederhana dan mudah untuk digabungkan dengan komponen lainnya. Functional component juga dapat menggunakan metode `useEffect()` untuk menangani perubahan state.

2) JSX

JSX (*JavaScript Syntax Extension*) adalah sintaks yang memungkinkan penulisan kode HTML di dalam JavaScript, digunakan untuk membangun komponen dan mengatur struktur dalam komponen React.

3) Virtual DOM (*Document Object Model*)

Virtual DOM adalah representasi yang lebih ringan dari DOM yang digunakan oleh ReactJS untuk meningkatkan performa aplikasi. Ketika ada perubahan pada state atau data, ReactJS tidak langsung mengubah DOM yang sebenarnya. Sebaliknya, ia memperbarui Virtual DOM terlebih dahulu, kemudian membandingkan perubahan yang terjadi dengan DOM yang lama. Proses ini dikenal sebagai "reconciliation" dan memungkinkan ReactJS untuk hanya memperbarui bagian yang perlu diubah, sehingga meningkatkan efisiensi rendering dan memberikan pengalaman pengguna yang lebih responsif.

4. PostgreSQL

a. Pengertian PostgreSQL

PostgreSQL, atau sering disebut Postgres, adalah salah satu sistem manajemen basis data relasional (RDBMS) yang populer dan banyak digunakan oleh berbagai instansi maupun perusahaan. Bersifat open source, PostgreSQL telah dikembangkan sejak lama dan awalnya dikenal dengan nama POSTGRES. Berbagai perusahaan ternama, seperti Instagram, HP, VMware, dan NASA, menggunakan PostgreSQL sebagai salah satu komponen teknologi mereka. (MF, 2020)

b. Fitur dan Penggunaan

PostgreSQL menyediakan fitur lengkap yang setara dengan DBMS lainnya, seperti SQL Server, Oracle, dan MySQL. Database ini dapat digunakan untuk:

- 1) OLTP (*Online Transactional Processing*): Digunakan untuk kebutuhan transaksi data.
- 2) OLAP (*Online Analytical Processing*): Digunakan untuk kebutuhan analisis data.

- 3) HTAP (*Hybrid Transactional/Analytical Processing*):
Mengombinasikan kebutuhan OLTP dan OLAP.

Dalam konteks ini, PostgreSQL akan digunakan terutama untuk mendukung kebutuhan OLTP.

5. Postman

Pengertian Postman menurut Adrianto (2024) adalah alat yang sangat populer di kalangan pengembang dan penguji perangkat lunak untuk melakukan pengujian API. Dengan antarmuka yang ramah pengguna, Postman memungkinkan pengguna untuk mengirim permintaan HTTP ke server dan menerima respons dengan mudah. Alat ini mendukung berbagai metode HTTP seperti GET, POST, PUT, dan DELETE, yang memungkinkan pengujian berbagai jenis operasi pada API.

Dalam konteks pengujian *load testing*, Postman berfungsi sebagai alat untuk mensimulasikan beban pengguna yang berbeda pada API. Dengan menggunakan Postman, penguji dapat mengatur sejumlah besar permintaan yang dikirim ke server dalam waktu yang bersamaan, sehingga dapat mengukur seberapa baik API dapat menangani beban tersebut. Hal ini penting untuk memastikan bahwa aplikasi dapat berfungsi dengan baik di bawah kondisi penggunaan yang tinggi.

Selain itu, Postman juga menyediakan fitur untuk mengelola koleksi permintaan, yang memungkinkan penguji untuk mengorganisir dan menjalankan serangkaian pengujian secara berurutan. Fitur ini sangat berguna dalam pengujian *load testing*, di mana penguji dapat dengan mudah mengubah parameter dan mengulangi pengujian untuk mendapatkan hasil yang konsisten.

Dengan kemampuan untuk mengukur metrik penting seperti waktu respons, throughput, dan tingkat kesalahan, Postman menjadi alat yang sangat berharga dalam proses pengujian. Hasil dari pengujian ini dapat memberikan wawasan yang mendalam tentang performa API dan

membantu pengembang dalam mengidentifikasi potensi masalah sebelum aplikasi diluncurkan ke pengguna akhir.

Secara keseluruhan, Postman tidak hanya mempermudah proses pengujian API, tetapi juga meningkatkan efisiensi dan efektivitas pengujian load testing, sehingga memastikan bahwa aplikasi yang dikembangkan dapat memenuhi kebutuhan pengguna dengan baik.

6. Visual Code Studio

Menurut Sri Hartati (2020), Visual Studio Code atau disingkat menjadi VS Code adalah sebuah editor kode sumber yang dikembangkan oleh Microsoft. Editor ini dirancang untuk mendukung berbagai bahasa pemrograman dan menyediakan berbagai fitur yang memudahkan pengembangan perangkat lunak. Dalam konteks pengembangan sistem informasi berbasis web, VS Code menjadi alat yang sangat berguna bagi para pengembang.

Salah satu keunggulan VS Code adalah kemampuannya untuk menjalankan program secara langsung, yang memungkinkan pengembang untuk menguji kode mereka dengan cepat. Selain itu, VS Code juga mendukung pembuatan file PHP, yang merupakan salah satu bahasa pemrograman yang umum digunakan dalam pengembangan aplikasi web. Dengan antarmuka yang intuitif dan dukungan untuk berbagai ekstensi, VS Code membantu pengembang dalam menyelesaikan tugas-tugas mereka dengan lebih efisien.

BAB III

ANALISIS DAN HASIL

A. Perencanaan Sistem

1. Identifikasi Kebutuhan

Langkah pertama adalah memahami kebutuhan sistem melalui analisis sederhana terhadap proses manual yang ada dan diskusi awal dengan pihak terkait. Beberapa kebutuhan utama yang ditentukan adalah:

- a. Pengguna utama sistem:
 - 1) Santri: Memerlukan sistem yang mudah digunakan untuk mengajukan izin keluar masuk secara digital.
 - 2) Penjaga Asrama: Membutuhkan akses untuk memproses pengajuan izin, termasuk melihat, menyetujui, atau menolak izin.
 - 3) Admin (Staf Administrasi): Bertugas mengelola data izin, memantau status pengajuan, dan membuat laporan sesuai kebutuhan.
- b. Fungsi utama sistem:
 - 1) Proses pengajuan izin dan persetujuan dilakukan secara digital.
 - 2) Data izin disimpan dan dicatat secara otomatis untuk akses yang lebih terorganisir.
 - 3) Tersedianya laporan yang dapat digunakan untuk evaluasi dan analisis.

2. Penentuan Ruang Lingkup Sistem

Ruang lingkup sistem difokuskan pada penggantian proses manual menjadi digital, dengan tujuan meningkatkan efisiensi dan kemudahan pengelolaan. Sistem akan mencakup:

- a. Digitalisasi proses pengajuan izin keluar masuk santri.
- b. Pemberitahuan otomatis kepada pihak terkait untuk mempercepat komunikasi.
- c. *Dashboard* untuk memantau status pengajuan izin secara langsung.

- d. Fitur laporan yang dapat diakses dengan mudah untuk keperluan pencatatan dan analisis.

3. Sumber Daya yang Diperlukan

Untuk mengembangkan Sistem Informasi Perizinan Keluar Masuk Santri, diperlukan berbagai sumber daya agar proses pengembangan dapat berjalan dengan lancar dan sistem yang dihasilkan dapat memenuhi kebutuhan pengguna. Sumber daya yang dibutuhkan meliputi:

- a. Perangkat Keras (*Hardware*)
 - 1) Server: Digunakan untuk menyimpan data dan menjalankan sistem berbasis web secara *online*.
 - 2) Komputer atau Laptop: Dipakai untuk pengembangan sistem dan pengelolaan data oleh admin.
 - 3) Perangkat Pengguna: Santri, penjaga asrama, dan admin memerlukan perangkat seperti komputer, laptop, atau ponsel pintar untuk mengakses sistem.
 - 4) Jaringan Internet: Diperlukan agar semua pengguna dapat terhubung ke sistem kapan saja dan di mana saja.
- b. Perangkat Lunak (*Software*)
 - 1) *Framework* Pengembangan: Spring Boot sebagai platform untuk membangun aplikasi berbasis web.
 - 2) *Database Management System* (DBMS): PostgreSQL digunakan untuk menyimpan dan mengelola data.
 - 3) *Text Editor* atau IDE: Visual Studio Code digunakan untuk menulis kode dan mengelola proyek pengembangan.
 - 4) Sistem Operasi: Windows digunakan sebagai platform utama pengembang dalam merancang dan mengembangkan sistem.
 - 5) *Dependency Management*: Maven untuk mengelola pustaka dan *framework* yang dibutuhkan dalam proyek Spring Boot.

B. Analisis Sistem

1. Sistem yang Sedang Berjalan

Saat ini, proses pengelolaan izin keluar dan masuk santri di Perguruan Diniyyah Puteri Padang Panjang masih dilakukan dengan cara manual. Setiap santri yang ingin keluar asrama harus mengisi formulir atau buku log yang mencatat informasi seperti alasan izin, tanggal, dan durasi izin. Setelah itu, petugas asrama akan memeriksa pengajuan tersebut dan memberikan keputusan, apakah izin disetujui atau ditolak, dengan menandatangani formulir tersebut. Proses ini memerlukan banyak waktu dan tenaga karena semuanya dilakukan secara manual.

Pemantauan izin yang telah diajukan juga dilakukan secara langsung. Setiap kali santri mengajukan izin, petugas harus memeriksa buku log untuk mengetahui status izin tersebut. Hal ini tentu saja membuat pemantauan lebih rumit dan memakan waktu, apalagi jika banyak pengajuan izin yang harus ditangani sekaligus. Proses manual ini membuat akses informasi menjadi lambat dan terkadang sulit, sehingga keputusan mengenai izin bisa terlambat.

Selain itu, laporan mengenai izin juga dibuat secara manual, dengan memeriksa catatan yang ada dalam buku atau formulir. Ini tentu saja memakan waktu lama dan rentan terhadap kesalahan, seperti data yang duplikat atau hilang, yang bisa mengganggu akurasi catatan izin.

Secara keseluruhan, sistem yang ada saat ini masih sangat bergantung pada proses manual, yang membuat pengelolaan izin menjadi lebih lama dan berisiko terhadap kesalahan. Proses ini menyita banyak waktu dan tenaga, serta sulit untuk diakses dan diperiksa dengan cepat. Oleh karena itu, sistem yang lebih terstruktur dan otomatis diperlukan untuk memperbaiki kelemahan-kelemahan ini, agar pengelolaan izin santri bisa berjalan dengan lebih mudah dan akurat.

2. Sistem yang Diusulkan

Sistem Informasi Perizinan Keluar Masuk Santri dirancang untuk mempermudah dan mempercepat proses pengajuan, persetujuan,

pencatatan, dan pemantauan izin di Asrama Perguruan Diniyyah Puteri Padang Panjang. Sistem berbasis web ini menggantikan cara manual yang selama ini dilakukan menggunakan formulir kertas dan buku catatan. Berikut alur kerja dari sistem yang diusulkan:

a. Pengajuan Izin oleh Santri

Ketika seorang santri ingin mengajukan izin keluar atau masuk, mereka dapat masuk ke dalam sistem menggunakan akun masing-masing. Setelah berhasil login, santri akan diarahkan ke menu pengajuan izin. Di menu ini, santri perlu mengisi formulir dengan data yang diperlukan, seperti alasan pengajuan, tanggal keluar, tanggal kembali, dan informasi tambahan lainnya. Setelah semua data lengkap, pengajuan tersebut dikirimkan melalui sistem untuk diproses lebih lanjut.

b. Pemberitahuan ke Penjaga Asrama

Setelah pengajuan dikirimkan oleh santri, sistem secara otomatis mengirimkan notifikasi kepada penjaga asrama yang bertugas. Notifikasi ini akan muncul di *dashboard* penjaga untuk memastikan mereka segera mengetahui adanya pengajuan baru yang membutuhkan persetujuan.

c. Peninjauan dan Persetujuan oleh Penjaga Asrama

Penjaga asrama dapat login ke sistem untuk memeriksa pengajuan izin yang masuk. Mereka akan melihat detail pengajuan, seperti alasan, jadwal keluar dan kembali, serta riwayat izin santri sebelumnya. Berdasarkan informasi tersebut, penjaga asrama dapat memutuskan apakah akan menyetujui atau menolak permohonan tersebut. Jika perlu, mereka juga dapat memberikan catatan tambahan sebelum mengambil keputusan.

d. Pemberitahuan Status kepada Santri

Setelah penjaga asrama membuat keputusan, sistem akan memberi tahu santri mengenai status permohonan mereka. Jika pengajuan disetujui, santri akan menerima rincian jadwal keluar dan

kembali beserta ketentuan yang harus dipatuhi. Jika ditolak, alasan penolakan akan disampaikan melalui sistem.

e. Pencatatan dan Pemantauan Izin

Semua pengajuan izin, baik yang disetujui maupun yang ditolak, akan tercatat secara otomatis dalam sistem. Data ini disimpan untuk mempermudah pemantauan dan pelaporan. Penjaga asrama serta admin dapat memantau status izin secara langsung melalui *dashboard*, termasuk izin yang aktif, selesai, atau yang ditolak.

f. Proses Keluar dan Masuk Santri

Pada hari pelaksanaan, santri yang telah mendapatkan izin harus melapor ke petugas yang bertugas. Petugas akan memverifikasi izin dengan mencocokkan data yang ada di sistem, seperti nomor izin atau informasi pribadi santri. Proses ini memastikan bahwa semua izin keluar dan masuk telah sesuai prosedur dan tercatat dengan baik.

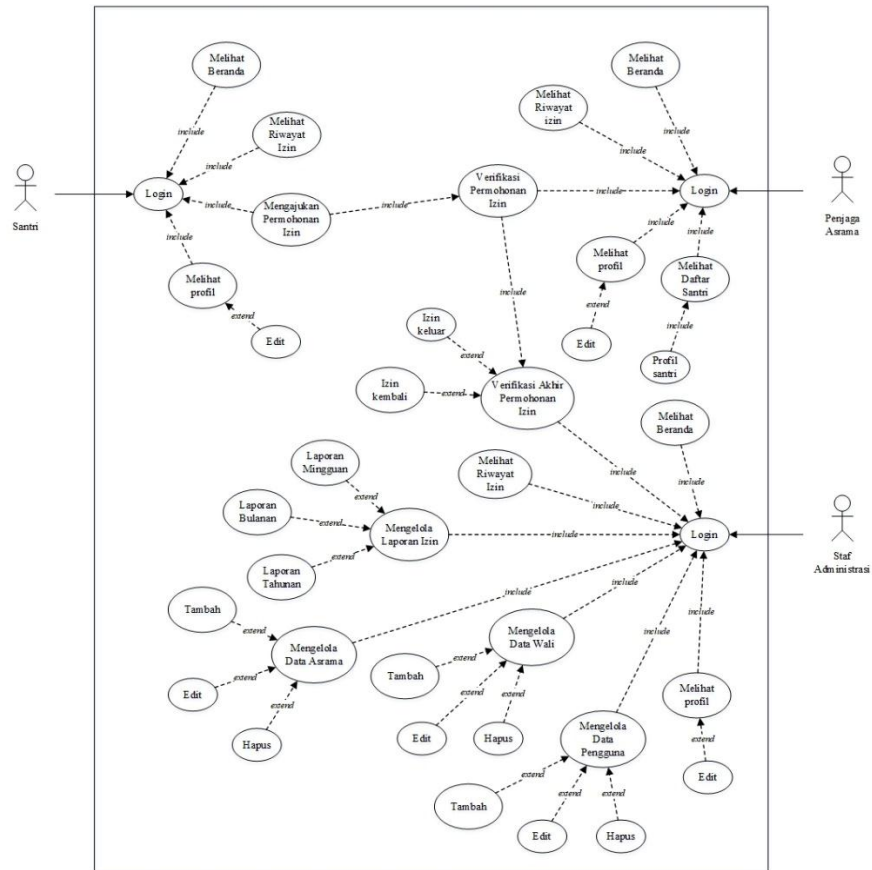
g. Laporan dan Analisis Data

Sistem menyediakan fitur untuk menghasilkan laporan secara otomatis berdasarkan data yang sudah tercatat. Laporan ini dapat dibuat secara harian, mingguan, atau bulanan, sesuai kebutuhan administrasi. Selain itu, data dalam laporan juga dapat digunakan untuk menganalisis pola izin keluar masuk santri, yang akan membantu pihak asrama dalam membuat keputusan atau kebijakan strategis.

C. Perancangan Sistem

1. Unified Modelling Language (UML)

a. Use Case Diagram



Gambar 3.1 Use Case Diagram

Diagram Use Case ini menunjukkan bagaimana pengguna berinteraksi dengan Sistem Informasi Perizinan Keluar Masuk Santri, yang dirancang untuk mempermudah proses perizinan. Terdapat tiga jenis pengguna: Santri, Penjaga Asrama, dan Staf Administrasi, masing-masing dengan peran dan akses yang berbeda.

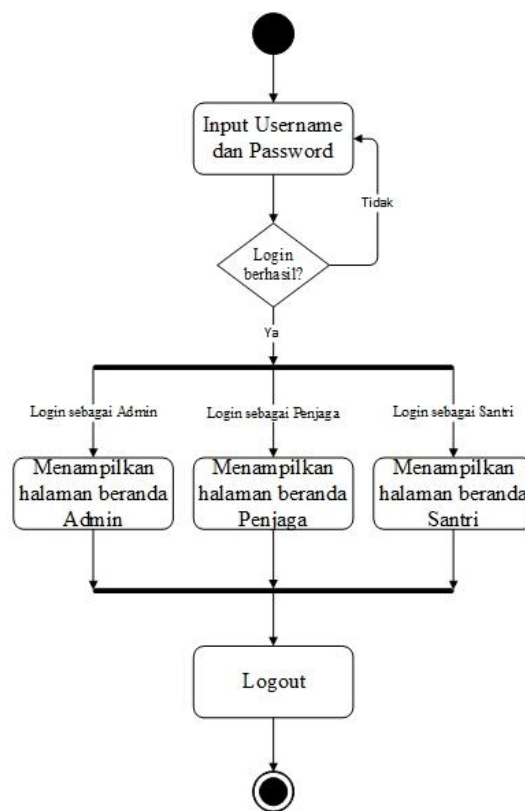
Santri dapat *login* untuk mengakses berbagai fitur, seperti melihat beranda, mengajukan permohonan izin, melacak riwayat izin, mengelola profil, dan melihat laporan mingguan atau bulanan. Permohonan izin santri akan diverifikasi oleh penjaga asrama dan staf administrasi.

Penjaga asrama bertugas memproses permohonan izin, memantau riwayat izin santri, dan mengakses informasi profil santri.

Sementara itu, staf administrasi memiliki tanggung jawab lebih luas, termasuk mengelola laporan izin, data asrama, data wali, dan pengguna sistem, serta mengatur profil mereka sendiri.

b. Activity Diagram

1) Login



Gambar 3.2 Activity Diagram Login

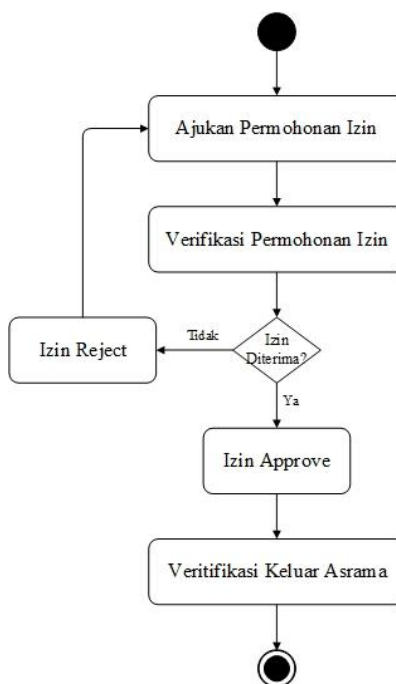
Diagram aktivitas di atas menunjukkan proses *login* dalam Sistem Informasi Perizinan Keluar Masuk Santri. Proses dimulai dengan pengguna memasukkan *username* dan *password* di halaman *login*. Sistem kemudian memverifikasi informasi tersebut untuk menentukan keberhasilan login.

Jika *login* gagal, pengguna diminta untuk memeriksa kembali data yang dimasukkan. Namun, jika berhasil, pengguna

akan diarahkan ke halaman beranda sesuai dengan jenis pengguna: Admin, Penjaga, atau Santri.

Setelah selesai, semua pengguna dapat keluar dari akun mereka dengan menekan tombol *logout*, yang mengakhiri sesi dan mengembalikan mereka ke halaman *login*.

2) Pengajuan Izin



Gambar 3.3 Activity Diagram Pengajuan Izin

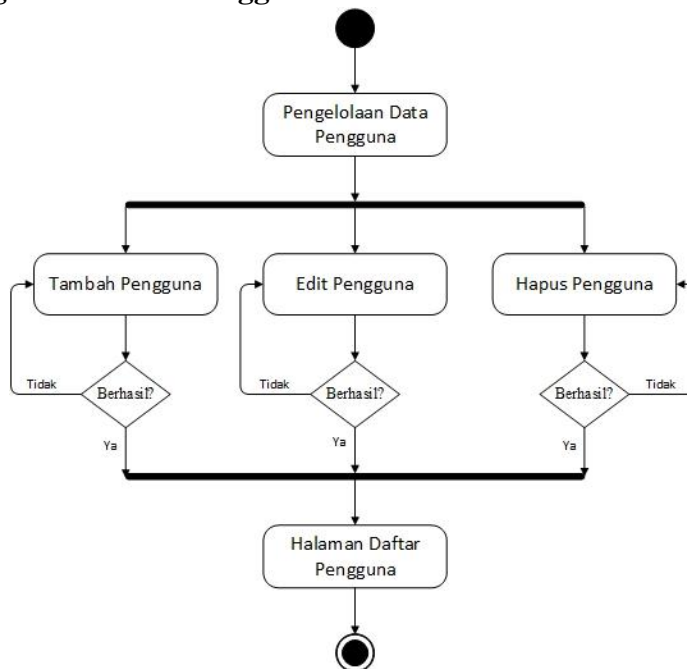
Proses pengajuan izin di asrama dimulai ketika seorang santri mengajukan permohonan izin. Setelah permohonan diinputkan, penjaga asrama akan memeriksa semua detail untuk memastikan bahwa informasi yang diberikan sudah lengkap dan sesuai dengan aturan yang ada.

Setelah proses verifikasi, ada dua kemungkinan hasil: jika permohonan tidak memenuhi syarat, santri akan diberitahu bahwa izin mereka ditolak. Namun, jika semua syarat terpenuhi, izin akan disetujui.

Pada langkah terakhir, sebelum meninggalkan asrama, santri akan menjalani verifikasi oleh Staf Administrasi sebagai

admin sistem untuk memastikan bahwa semua prosedur telah diikuti dengan baik.

3) Pengelolaan Data Pengguna



Gambar 3.4 Activity Diagram Pengelolaan Data Pengguna

Dalam pengelolaan data pengguna, staf administrasi memulai dari halaman utama untuk memilih tindakan yang ingin dilakukan, seperti menambah, mengedit, atau menghapus data.

Ketika menambah pengguna baru, admin mengisi formulir yang ada. Jika penambahan berhasil, mereka akan kembali ke halaman daftar pengguna yang sudah diperbarui. Jika ada kesalahan, admin bisa memperbaiki dan mencoba lagi.

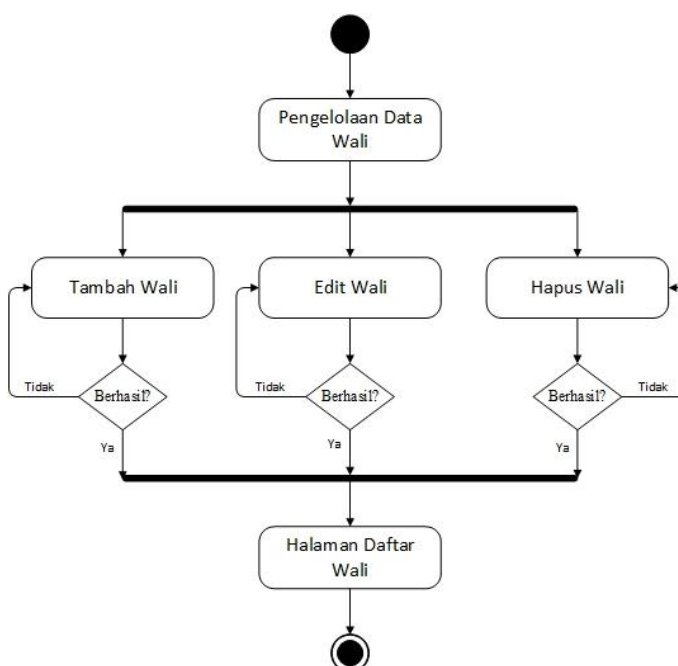
Untuk mengedit data yang sudah ada, admin memperbarui informasi yang diperlukan. Setelah itu, sistem akan memeriksa apakah perubahan berhasil. Jika ya, admin akan melihat data yang diperbarui; jika tidak, mereka bisa memperbaiki masalah tersebut.

Admin juga bisa menghapus data yang tidak diperlukan. Setelah memilih data untuk dihapus, sistem akan mengonfirmasi

apakah penghapusan berhasil. Jika berhasil, data akan hilang dari daftar, dan admin akan melihat tampilan terbaru.

Setiap kali ada perubahan, admin akan kembali ke halaman daftar pengguna, yang selalu menampilkan informasi terbaru. Proses ini memastikan bahwa pengelolaan data pengguna berjalan dengan baik dan teratur.

4) Pengelolaan Data Wali



Gambar 3.5 Activity Diagram Pengelolaan Data Wali

Dalam pengelolaan data wali, staf administrasi memulai dari halaman utama untuk memilih tindakan yang ingin dilakukan, seperti menambah, mengedit, atau menghapus data.

Ketika menambah wali baru, admin mengisi formulir yang ada. Jika penambahan berhasil, mereka akan kembali ke halaman daftar wali yang sudah diperbarui. Jika ada kesalahan, admin bisa memperbaiki dan mencoba lagi.

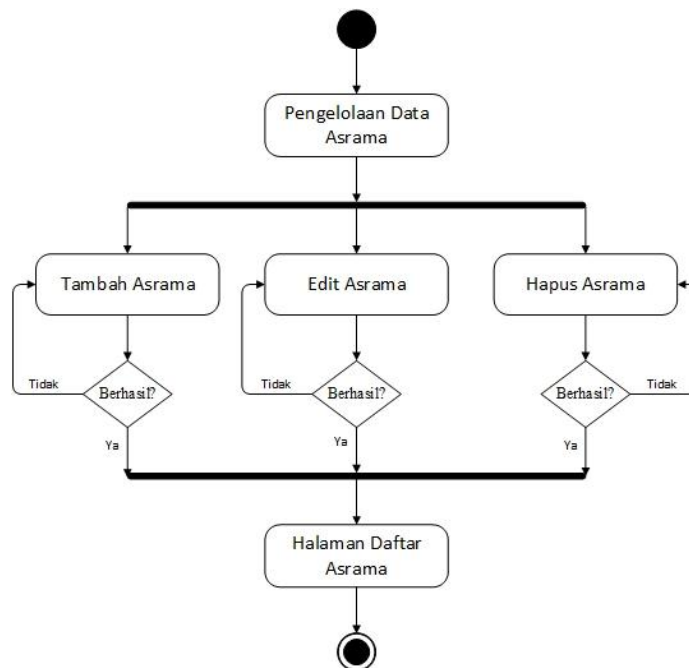
Untuk mengedit data yang sudah ada, admin memperbarui informasi yang diperlukan. Setelah itu, sistem akan memeriksa

apakah perubahan berhasil. Jika ya, admin akan melihat data yang diperbarui; jika tidak, mereka bisa memperbaiki masalah tersebut.

Admin juga bisa menghapus data yang tidak diperlukan. Setelah memilih data untuk dihapus, sistem akan mengonfirmasi apakah penghapusan berhasil. Jika berhasil, data akan hilang dari daftar, dan admin akan melihat tampilan terbaru.

Setiap kali ada perubahan, admin akan kembali ke halaman daftar wali, yang selalu menampilkan informasi terbaru. Proses ini memastikan bahwa pengelolaan data wali berjalan dengan baik dan teratur.

5) Pengelolaan Data Asrama



Gambar 3.6 Activity Diagram Pengelolaan Data Asrama

Dalam pengelolaan data asrama, staf administrasi memulai dari halaman utama untuk memilih tindakan yang ingin dilakukan, seperti menambah, mengedit, atau menghapus data.

Ketika menambah asrama baru, admin mengisi formulir yang ada. Jika penambahan berhasil, mereka akan kembali ke

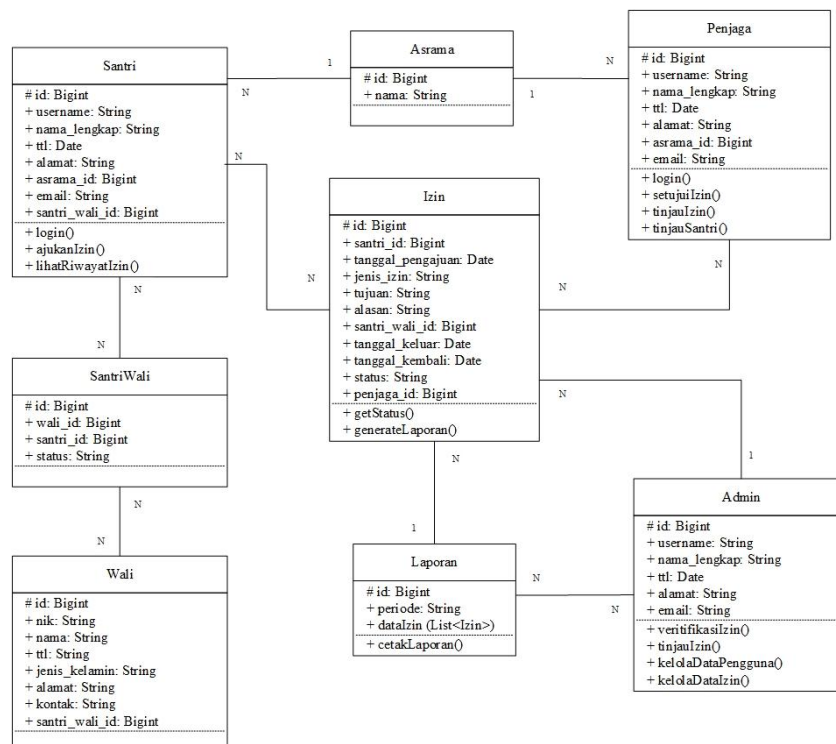
halaman daftar asrama yang sudah diperbarui. Jika ada kesalahan, admin bisa memperbaiki dan mencoba lagi.

Untuk mengedit data yang sudah ada, admin memperbarui informasi yang diperlukan. Setelah itu, sistem akan memeriksa apakah perubahan berhasil. Jika ya, admin akan melihat data yang diperbarui; jika tidak, mereka bisa memperbaiki masalah tersebut.

Admin juga bisa menghapus data yang tidak diperlukan. Setelah memilih data untuk dihapus, sistem akan mengonfirmasi apakah penghapusan berhasil. Jika berhasil, data akan hilang dari daftar, dan admin akan melihat tampilan terbaru.

Setiap kali ada perubahan, admin akan kembali ke halaman daftar asrama, yang selalu menampilkan informasi terbaru. Proses ini memastikan bahwa pengelolaan data asrama berjalan dengan baik dan teratur.

c. *Class Diagram*



Gambar 3.7 *Class Diagram*

Diagram ini menunjukkan bagaimana sistem mengelola data santri, wali, izin, laporan, dan peran pengguna lain seperti penjaga asrama dan admin.

Santri terhubung dengan asrama tempat mereka tinggal, serta memiliki hubungan dengan wali yang dikelola melalui tabel relasi SantriWali, yang juga mencatat status hubungan tersebut. Ketika santri mengajukan izin, data mereka tercatat dalam tabel Izin, termasuk informasi seperti tanggal pengajuan, alasan, tujuan, dan status izin.

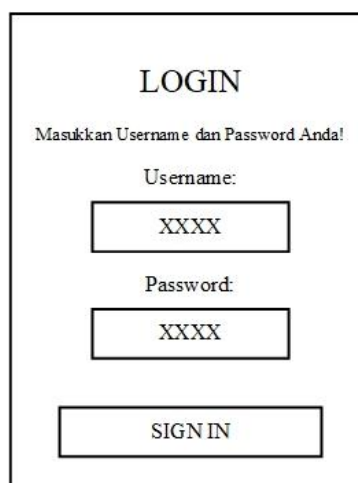
Penjaga bertugas meninjau dan menyetujui izin yang diajukan santri, sementara admin memiliki tanggung jawab lebih luas, termasuk mengelola data pengguna, izin, dan membuat laporan. Laporan dirancang untuk mengelompokkan data izin berdasarkan periode tertentu, membantu pemantauan dan evaluasi.

2. Antarmuka (*Interface*)

a. Desain *Input*

1) Tampilan Menu *Login*

Formulir menu *login* ditampilkan pada halaman ini. Desain lengkap menu *login* ditunjukkan pada gambar berikut ini:



The image shows a login form with the following elements:

- Title: LOGIN
- Instruction: Masukkan Username dan Password Anda!
- Label: Username:
- Input field: XXXX
- Label: Password:
- Input field: XXXX
- Button: SIGN IN

Gambar 3.8 Tampilan Menu *Login*

2) Tampilan Permohonan Izin

Formulir permohonan izin ditampilkan pada halaman ini. Berikut ini merupakan desain lengkap dari formulir tersebut:

Permohonan Izin

Jenis Izin:

Tujuan:

Alasan:

Wali Penjemput:

Tgl Keluar:

Tgl Kembali:

Gambar 3.9 Tampilan Pengajuan Permohonan Izin

3) Tampilan Verifikasi Permohonan Izin

Formulir verifikasi permohonan izin ditampilkan pada halaman ini. Berikut ini merupakan desain lengkap dari formulir tersebut:

Verifikasi Permohonan Izin

No	Jenis Izin	Tujuan	Alasan	Wali Penjemput	Tgl Keluar	Tgl Kembali	Aksi
99	XXXX	XXXXXX	XXXXXX	XXXXXX	99/99/9999	99/99/9999	<input type="button" value="APPROVE"/> <input type="button" value="REJECT"/>
99	XXXX	XXXXXX	XXXXXX	XXXXXX	99/99/9999	99/99/9999	<input type="button" value="APPROVE"/> <input type="button" value="REJECT"/>

Gambar 3.10 Tampilan Verifikasi Permohonan Izin

4) Tampilan Verifikasi Izin Keluar

Formulir verifikasi izin keluar dari asrama ditampilkan pada halaman ini. Berikut ini merupakan desain lengkap dari formulir tersebut:

Verifikasi Izin Keluar										
No	Nama Santri	Jenis Izin	Tujuan	Alasan	Wali Penjemput	Tgl Keluar	Tgl Kembali	Status Izin	Nama Penjaga	Aksi
99	XXXX	XXXX	XXXX	XXXX	XXXXXX	99/99/9999	99/99/9999	APPROVE	XXXXXX	Tandai Keluar
99	XXXX	XXXX	XXXX	XXXX	XXXXXX	99/99/9999	99/99/9999	APPROVE	XXXXXX	Tandai Keluar

Gambar 3.11 Tampilan Verifikasi Izin Keluar

5) Tampilan Verifikasi Izin Kembali

Formulir verifikasi izin kembali ke asrama ditampilkan pada halaman ini. Berikut ini merupakan desain lengkap dari formulir tersebut:

Verifikasi Izin Kembali										
No	Nama Santri	Jenis Izin	Tujuan	Alasan	Wali Penjemput	Tgl Keluar	Tgl Kembali	Status Izin	Nama Penjaga	Aksi
99	XXXX	XXXX	XXXX	XXXX	XXXXXX	99/99/9999	99/99/9999	APPROVE	XXXXXX	Tandai Kembali
99	XXXX	XXXX	XXXX	XXXX	XXXXXX	99/99/9999	99/99/9999	APPROVE	XXXXXX	Tandai Kembali

Gambar 3.12 Tampilan Verifikasi Izin Kembali

6) Tampilan Penambahan Data Santri

Formulir penambahan data santri ditampilkan pada halaman ini. Berikut ini merupakan desain lengkap dari formulir tersebut:

Tambah Data Santri

Username :

Password :

Nama Lengkap:

TTL :

Email :

Asrama : ▾

Alamat :

Foto Profil : No file choosen

Gambar 3.13 Tampilan Penambahan Data Santri

7) Tampilan Penambahan Data Penjaga

Formulir penambahan data penjaga ditampilkan pada halaman ini. Berikut ini merupakan desain lengkap dari formulir tersebut:

Tambah Data Penjaga

Username :

Password :

Nama Lengkap:

TTL :

Email :

Asrama : ▾

Alamat :

Foto Profil : No file choosen

Gambar 3.14 Tampilan Penambahan Data Penjaga

8) Tampilan Penambahan Data Wali

Formulir penambahan data asrama ditampilkan pada halaman ini. Berikut ini merupakan desain lengkap dari formulir tersebut

Tambah Data Wali

NIK	:	<input style="width: 95%;" type="text" value="9999999999999999"/>
Nama	:	<input style="width: 95%;" type="text" value="XXXXXX"/>
TTL	:	<input style="width: 95%;" type="text" value="XXXXXXXXXX"/>
Jenis Kelamin	:	<input style="border-bottom: none; border-top: none; border-left: none; border-right: none;" type="text" value="Pilih Jenis Kelamin"/> ▾
Alamat	:	<input style="width: 95%;" type="text" value="XXXXXXXXXX"/>
Alamat	:	<input style="width: 95%;" type="text" value="XXXXXXXXXX"/>
Kontak	:	<input style="width: 95%;" type="text" value="999999999999"/>
Username Santri	:	<input style="width: 95%;" type="text" value="XXXXXXXXXX"/>
Status	:	<input style="border-bottom: none; border-top: none; border-left: none; border-right: none;" type="text" value="Pilih Status"/> ▾

Gambar 3.15 Tampilan Penambahan Data Wali

9) Tampilan Penambahan Data Asrama

Formulir penambahan data asrama ditampilkan pada halaman ini. Berikut ini merupakan desain lengkap dari formulir tersebut:

Tambah Data Asrama

Nama Asrama:	<input style="width: 95%;" type="text" value="XXXXXX"/>
--------------	---

Gambar 3.16 Tampilan Penambahan Data Asrama

10) Tampilan Pengelolaan Data SantriWali

Formulir pengelolaan data hubungan antara santri dan wali ditampilkan pada halaman ini. Berikut ini merupakan desain lengkap dari formulir tersebut:

Kelola Data Santri Wali

No.	Santri	Status Sebagai	Aksi
Z	Z	Z	<input type="button" value="Hapus"/>

Username Santri :

Status :

Gambar 3.17 Tampilan Pengelolaan Data SantriWali

b. Desain Output

1) Tampilan Riwayat Izin

Tampilan Riwayat Izin ini berfungsi untuk melihat informasi data perizinan. Laporan ini menampilkan nama santri, jenis izin, tujuan, alasan, wali yang menjemput, tanggal keluar asrama, tanggal kembali ke asrama, nama penjaga asrama yang memproses izin, santri sudah keluar dari asrama, santri sudah kembali ke asrama, dan keterangan. Adapun bentuk dari desain dapat dilihat dari gambar berikut:

Riwayat Izin

Mingguan/Bulanan/Tahunan
Tanggal/Bulan/Tahun: XXXX

No	Nama Santri	Jenis Izin	Tujuan	Alasan	Wali Penjemput	Tgl Keluar	Tgl Kembali	Status Izin	Nama Penjaga	Keluar	Kembali	Keterangan
Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z	Z

Gambar 3.18 Tampilan Riwayat Izin

2) Tampilan Profil Santri

Tampilan Profil Santri ini berfungsi untuk melihat informasi data santri. Profil ini menampilkan *username*, *password*, nama lengkap, tempat dan tanggal lahir, email, asrama, alamat, nama wali, serta kontak wali dari santri tersebut. Adapun bentuk dari desain dapat dilihat dari gambar berikut:

Profil Santri		
Foto Profil	Username	: XXXXX
	Password	: XXXXX
	Nama Lengkap	: XXXXXXXXXXXXX
	TTL	: XXXXX, 99/99/9999
	Email	: XXXXXXXX
	Asrama	: XXXXXX
	Alamat	: XXXXXXXXXXXXXXXXXXXXX
	Nama Wali	: XXXXXXXXX
	No Tlp Wali	: 999999999

Gambar 3.19 Tampilan Profil Santri

3) Tampilan Profil Penjaga

Tampilan Profil Penjaga ini berfungsi untuk melihat informasi data penjaga. Profil ini menampilkan *username*, *password*, nama lengkap, tempat dan tanggal lahir, email, asrama, serta alamat dari penjaga tersebut. Adapun bentuk dari desain dapat dilihat dari gambar berikut:

Profil Penjaga		
Foto Profil	Username	: XXXXX
	Password	: XXXXX
	Nama Lengkap	: XXXXXXXXXXXXX
	TTL	: XXXXX, 99/99/9999
	Email	: XXXXXXXX
	Asrama	: XXXXXX
	Alamat	: XXXXXXXXXXXXXXXXXXXXX

Gambar 3.20 Tampilan Profil Penjaga

4) Tampilan Profil Admin

Tampilan Profil Admin ini berfungsi untuk melihat informasi data admin. Profil ini menampilkan *username*, *password*, nama lengkap, tempat dan tanggal lahir, email, serta alamat dari penjaga tersebut. Adapun bentuk dari desain dapat dilihat dari gambar berikut:

Profil Admin		
Foto Profil	Username	: XXXXX
	Password	: XXXXX
	Nama Lengkap	: XXXXXXXXXXXXX
	TTL	: XXXXX, 99/99/9999
	Email	: XXXXXXX
	Alamat	: XXXXXXXXXXXXXXXXXXXXX

Gambar 3.21 Tampilan Profil Admin

5) Tampilan Profil Wali

Tampilan Profil Wali ini berfungsi untuk melihat informasi data wali. Profil ini menampilkan NIK, nama, tempat dan tanggal lahir, kontak, serta alamat dari wali tersebut. Adapun bentuk dari desain dapat dilihat dari gambar berikut:

Profil Wali		
NIK	:	9999999999999999
Nama	:	XXXXX
TTL	:	XXXXX, 99/99/9999
Jenis Kelamin	:	XXXXXXXX
Alamat	:	XXXXXXXXXXXXXXXXXX
Kontak	:	999999999999
Santri	:	XXXXXX sebagai XXXXXX

Gambar 3.22 Tampilan Profil Wali

6) Tampilan Daftar Santri

Tampilan Daftar Santri ini berfungsi untuk melihat informasi keseluruhan data santri. Profil ini menampilkan *username*, nama lengkap, tempat dan tanggal lahir, email, asrama, alamat, nama wali, serta kontak wali dari santri tersebut. Adapun bentuk dari desain dapat dilihat dari gambar berikut:

Daftar Santri									
No	Username	Nama Lengkap	TTL	Email	Asrama	Alamat	Nama Wali	No Tlp Wali	Aksi
Z	Z	Z	Z	Z	Z	Z	Z	Z	<input type="button" value="+ Tambah"/> <input type="button" value="Edit"/> <input type="button" value="Hapus"/>

Gambar 3.23 Tampilan Daftar Santri

7) Tampilan Daftar Penjaga

Tampilan Daftar Penjaga ini berfungsi untuk melihat informasi keseluruhan data penjaga. Profil ini menampilkan *username*, nama lengkap, tempat dan tanggal lahir, email, asrama, serta alamat dari penjaga tersebut. Adapun bentuk dari desain dapat dilihat dari gambar berikut:

Daftar Penjaga							
No	Username	Nama Lengkap	TTL	Email	Asrama	Alamat	Aksi
Z	Z	Z	Z	Z	Z	Z	<input type="button" value="+ Tambah"/> <input type="button" value="Edit"/> <input type="button" value="Hapus"/>

Gambar 3.24 Tampilan Daftar Penjaga

8) Tampilan Daftar Wali

Tampilan Daftar Wali ini berfungsi untuk melihat informasi keseluruhan data wali. Profil ini menampilkan NIK,

nama, tempat dan tanggal lahir, alamat, kontak, serta santri yang memiliki hubungan dengan wali tersebut. Adapun bentuk dari desain dapat dilihat dari gambar berikut:

Daftar Wali								
No	NIK	Nama	TTL	Jenis Kelamin	Alamat	Kontak	Santri	Aksi
Z	Z	Z	Z	Z	Z	Z	Z	<input type="button" value="+ Tambah"/> <input type="button" value="Edit"/> <input type="button" value="Hapus"/>

Gambar 3.25 Tampilan Daftar Wali

9) Tampilan Daftar Asrama

Tampilan Daftar Asrama ini berfungsi untuk melihat informasi keseluruhan data asrama. Profil ini menampilkan nama asrama tersebut. Adapun bentuk dari desain dapat dilihat dari gambar berikut:

Daftar Asrama		
No	Alamat	Aksi
Z	Z	<input type="button" value="+ Tambah"/> <input type="button" value="Edit"/> <input type="button" value="Hapus"/>

Gambar 3.26 Tampilan Daftar Asrama

10) Tampilan Laporan Izin

Tampilan Laporan Izin ini berfungsi untuk laporan dari perizinan. Laporan ini menampilkan total izin, total izin dengan status disetujui dan ditolak, serta total santri yang sudah keluar dari asrama dan kembali ke asrama. Pada halaman tersebut juga

menampilkan daftar riwayat izin. Adapun bentuk dari desain dapat dilihat dari gambar berikut:

Laporan Izin

Laporan Mingguan/Bulanan/Tahunan
Tanggal/Bulan/Tahun: XXXX

Total : 999
Total status approve : 999
Total status reject : 999
Total sudah keluar : 999
Total sudah kembali : 999

No	Nama Santri	Jenis Izin	Tujuan	Alasan	Wali Penjemput	Tgl Keluar	Tgl Kembali	Status Izin	Nama Penjaga	Keluar	Kembali	Keterangan
1	2	3	4	5	6	7	8	9	10	11	12	13

Gambar 3.27 Tampilan Laporan Izin

c. Desain Database

Desain *database* ini dibuat untuk mendukung pengelolaan sistem dengan lebih mudah dan terstruktur. Ada tiga tabel utama dalam database ini, yaitu Pengguna, Izin, dan Asrama, yang saling terhubung sehingga data dapat diintegrasikan dengan baik. Berikut ini merupakan desain tabel-tabel tersebut:

1) Tabel Pengguna

Database : perizinan

Field Key : id

Fungsi : Menyimpan data pengguna

Tabel 3.1 Tabel Pengguna

No	Nama Field	Type	Width	Deskripsi
1	id	Bigint	-	Id pengguna
2	username	Varchar	255	Username pengguna
3	password	Varchar	255	Password pengguna
4	nama_lengkap	Varchar	255	Nama Pengguna
5	ttl	Varchar	255	Tempat dan tanggal

				lahir pengguna
6	email	Varchar	255	Email pengguna
7	asrama_id	Bigint	-	Asrama pengguna
8	alamat	Varchar	255	Alamat pengguna
9	santri_wali_id	Bigint	-	Wali dari pengguna santri
11	profile_image	Varchar	255	Foto profil pengguna
12	role	Varchar	255	Peran pengguna (Admin, Penjaga, atau Santri)

2) Tabel Wali

Database : perizinan

Field Key : id

Fungsi : Menyimpan data wali

Tabel 3.2 Tabel Wali

No	Nama Field	Type	Width	Deskripsi
1	id	Bigint	-	Id wali
2	nik	Varchar	255	NIK wali
3	nama	Varchar	255	Nama wali
4	ttl	Varchar	255	Tempat dan tanggal lahir wali
5	jenis_kelamin	Varchar	255	Jenis kelamin wali
6	alamat	Varchar	255	Alamat wali
7	kontak	Varchar	255	Kontak wali
8	santri_wali_id	Bigint	-	Santri yang berhubungan dengan wali

3) Tabel SantriWali

Database : perizinan*Field Key* : id

Fungsi : Menyimpan data hubungan antara wali dan santri

Tabel 3.3 Tabel SantriWali

No	Nama Field	Type	Width	Deskripsi
1	id	Bigint	-	Id SantriWali
2	wali_id	Varchar	255	Id Wali
3	santri_id	Varchar	255	Id Santri
4	status	Varchar	255	Status hubungan wali dengan santri

4) Tabel Asrama

Database : perizinan*Field Key* : id

Fungsi : Menyimpan data asrama

Tabel 3.4 Tabel Asrama

No	Nama Field	Type	Width	Deskripsi
1	id	Bigint	-	Id asrama
2	nama	Varchar	255	Nama asrama

5) Tabel Izin

Database : perizinan*Field Key* : id

Fungsi : Menyimpan data izin

Tabel 3.5 Tabel Izin

No	Nama Field	Type	Width	Deskripsi
1	id	Bigint	-	Id izin

2	santri_id	Bigint	-	Santri yang mengajukan izin
3	jenis_izin	Smallint	-	Jenis Izin
4	tujuan	Varchar	255	Tujuan atau lokasi
5	alasan	Varchar	255	Alasan izin
6	santri_wali_id	Bigint	-	Wali yang menjemput
7	waktu_pengajuan	Date	-	Waktu pengajuan izin
8	tanggal_keluar	Date	-	Tanggal keluar dari asrama
9	tanggal_kembali	Date	-	Tanggal kembali ke asrama
10	status	Varchar	255	Status izin
11	penjaga_id	Bigint	-	Penjaga yang memproses pengajuan izin
12	sudah_keluar	Boolean	-	Status keluar dari asrama
13	sudah_kembali	Boolean	-	Status kembali ke asrama
14	keterangan	Varchar	255	Keterangan

BAB IV

PENUTUP

A. KESIMPULAN

Dengan adanya sistem informasi perizinan yang diusulkan, pengelolaan izin keluar dan masuk asrama di Perguruan Diniyyah Puteri Padang Panjang akan mengalami beberapa perbaikan yang menjawab permasalahan yang telah diidentifikasi:

1. Meningkatkan Kemudahan dalam Proses: Sistem ini akan menghapus kebutuhan untuk mengelola izin secara manual, sehingga mengurangi waktu dan tenaga yang diperlukan dalam proses pengajuan dan pemrosesan izin.
2. Mengurangi Kesalahan Data: Dengan adanya sistem digital, risiko kesalahan seperti kehilangan data, duplikasi informasi, dan pemalsuan data dapat diminimalkan. Hal ini akan membuat catatan perizinan santri menjadi lebih akurat dan dapat diandalkan.
3. Mempermudah Pemantauan dan Pengambilan Keputusan: Sistem informasi ini akan memudahkan pemantauan izin dan mempercepat pengambilan keputusan oleh staf administrasi, sehingga proses perizinan dapat berjalan lebih lancar dan tepat waktu.

Dengan demikian, penerapan sistem informasi perizinan ini diharapkan dapat mengatasi masalah yang ada dan meningkatkan kualitas pengelolaan izin di lingkungan pesantren.

B. SARAN

Berdasarkan hasil penelitian dan kesimpulan yang telah diuraikan, berikut adalah beberapa saran yang dapat dipertimbangkan untuk meningkatkan pengelolaan izin:

1. Sistem dapat dikembangkan lebih jauh dengan menambahkan fitur notifikasi berbasis SMS atau email agar santri dan wali mendapatkan pemberitahuan otomatis terkait status izin.

2. Implementasi metode keamanan tambahan, seperti enkripsi data atau autentikasi dua faktor, perlu dipertimbangkan untuk melindungi informasi pengguna.
3. Untuk memastikan sistem ini berjalan dengan baik, perlu diadakan pelatihan bagi staf administrasi, petugas asrama, dan santri, serta disediakan panduan penggunaan. Pelatihan ini bertujuan agar mereka memahami cara kerja sistem dan dapat menggunakannya dengan benar.
4. Seiring waktu, sistem ini dapat dikembangkan lebih lanjut dengan menambahkan fitur yang lebih lengkap. Misalnya, menghubungkannya dengan sistem lain di Perguruan Diniyyah Puteri, seperti sistem akademik atau keuangan, agar manajemen menjadi lebih terintegrasi.

DAFTAR PUSTAKA

- Andrianto, L. D., & Suyatno, D. F. (2024). Analisis Performa Load Testing Antara Mysql Dan Nosql Mongoddb Pada RestAPI Nodejs Menggunakan Postman. *Journal of Emerging Information System and Business Intelligence (JEISBI)*, 5(1), 18–26. <https://ejournal.unesa.ac.id/index.php/JEISBI/article/view/58157%0Ahttps://ejournal.unesa.ac.id>
- Badan Pengembangan dan Pembinaan Bahasa. (2023). *Kamus Besar Bahasa Indonesia* (6th ed.). Balai Pustaka. <https://kbbi.kemdikbud.go.id/>
- Broadcom. (2024). *Spring: The source for modern Java development*. <https://spring.io/>
- Dwipa, R., Oktarina, S., Teisnajaya, U., & Jaka Pangestu, M. (2023). Impelementasi Sistem Qrcode Card Sebagai Control Keluar Masuk Santri Berbasis Web (Studi Kasus Pesantren Qodratullah). *Klik - Jurnal Ilmu Komputer*, 4(1), 42–49. <https://doi.org/10.56869/klik.v4i1.514>
- Khaqiqi, M. I. T., & Harani, N. H. (2023). *Penerapan Metode Gamifikasi Pada Rest Api Spring Boot*. Penerbit Buku Pedia. https://books.google.co.id/books?id=t76-EAAAQBAJ&newbks=0&printsec=frontcover&pg=PA30&dq=penerapan+metode+gamifikasi+pada+rest+api+spring+boot&hl=id&source=newbks_fb&redir_esc=y#v=onepage&q=penerapan+metode+gamifikasi+pada+rest+api+spring+bo
- Kusnawi, Romadhon, W., & Duhita, W. M. P. (2024). *Menyelami Dunia React.JS: Best Practice Berdasarkan Pengalaman di Industri IT*. Penerbit Andi. https://books.google.co.id/books?id=WuArEQAAQBAJ&newbks=1&newbks_redir=0&hl=id&redir_esc=y
- Marlindawati. (2022). *Penerapan EOQ (Economic Order Quantity) pada Sistem Informasi Investor*. <http://eprints.binadarma.ac.id/id/eprint/16284>
- MF, R. Y. (2020). *Pemrograman SQL dan PL/pgSQL di Database PostgreSQL Native Enterprise*. <https://books.google.co.id/books?id=I4HgDwAAQBAJ&newbks=0&printsec=frontcover&hl=id#v=onepage&q&f=false>
- Perguruan Diniyyah Puteri Padang Panjang. (2024). *Perguruan Diniyyah Puteri Padang Panjang*. <https://diniyyahputeri.org>
- Prehanto, D. R. (2020). *BUKU AJAR KONSEP SISTEM INFORMASI*. Scopindo Media Pustaka. https://books.google.co.id/books?id=0OriDwAAQBAJ&newbks=1&newbks_redir=0&printsec=frontcover&pg=PR1&dq=Buku+ajar+konsep+sistem+informasi.&hl=id&redir_esc=y#v=onepage&q=Buku+ajar+konsep+sistem

informasi.&f=false

- Pudyatmoko, Y. S. (2024). *PERIZINAN - Problem dan Upaya Pembinaan*. Penerbit Andi.
https://books.google.co.id/books?id=i4n9EAAAQBAJ&pg=PA20&dq=sistem+perizinan&hl=id&newbks=1&newbks_redir=0&sa=X&ved=2ahUKEwjA08SDqaKKAxWIZjgGHdA-LnMQ6AF6BAgJEAI#v=onepage&q=sistem+perizinan&f=false
- Ridwan, M., Fitri, I., & Benrahman, B. (2021). Rancang Bangun Marketplace Berbasis Website menggunakan Metodologi Systems Development Life Cycle (SDLC) dengan Model Waterfall. *Jurnal JTIK (Jurnal Teknologi Informasi Dan Komunikasi)*, 5(2), 173. <https://doi.org/10.35870/jtik.v5i2.209>
- Sri Hartati. (2020). Perancangan Sistem Informasi Inventaris Barang Pada Kantor Notaris Dan Ppat Ra Lia Kholila, Sh Menggunakan Visual Studio Code. *Siskomti*, 2(2), 37–48. <https://ejournal.uniled.ac.id/index.php/ITBis-SISKOMTI/article/view/123>
- The PostgreSQL Global Development Group. (2024). *PostgreSQL documentation*. <https://www.postgresql.org/docs/>
- Warno. (2020). Pembelajaran Pemrograman Bahasa Java Dan Arti Keyword. *Pembelajaran Pemrograman Bahasa Java Dan Arti Keyword*, 8(1), 40–51. <https://ejournal.esaunggul.ac.id/index.php/JIK/article/view/479/444>

