

---

# SISTEM INFORMASI PENGAJUAN CUTI, IZIN, DAN LEMBUR KARYAWAN MENGGUNAKAN *FRAMEWORK* ASP.NET MVC MELALUI PENDEKATAN *SCRUMBAN*

Nadya Fimatul Fadilah<sup>1)</sup>, Yayak Kartika Sari<sup>2)</sup>, Taufiq Agung Cahyono<sup>3)</sup>

<sup>1)</sup>Program Studi Informatika, Fakultas Sains dan Teknologi, Universitas Bhinneka PGRI  
Jl. Mayor Sujadi No.7, Manggis, Plosokandang, Kec. Kedungwaru, Kabupaten Tulungagung,  
Jawa Timur 66229

e-mail: [nadya.fimatul.fadilah@gmail.com](mailto:nadya.fimatul.fadilah@gmail.com)<sup>1)</sup>

\*corresponding author

(Naskah masuk : 17 Juni 2025 Diterima untuk diterbitkan : 10 September 2025)

## ABSTRAK

Salah satu divisi penting dalam Rumah Sakit Era Medika adalah divisi Human Resource Departemen (HRD) yang memajemen sumber daya manusia diantaranya adalah pengelolaan cuti, lembur, dan izin karyawan. Namun pada prosesnya, pengajuan ini masih dilakukan secara manual dengan menggunakan formulir kertas yang selanjutnya akan diajukan kepada atasan yang terkait. Data yang ada selanjutnya dimasukkan kedalam excel untuk dijadikan laporan. Hal ini beresiko menimbulkan berbagai kendala seperti keterlambatan dalam proses persetujuan, kehilangan data, dan kurangnya efisiensi. Penelitian ini bertujuan untuk mengembangkan sistem informasi pengajuan cuti, izin, dan lembur berbasis website guna mempermudah pengajuan dan pelaporan. Pengembangan ini dilakukan dengan menggunakan framework ASP.NET MVC yang dapat mengoptimalkan proses pengajuan dan persetujuan menjadi lebih efisien dan terstruktur. Pengumpulan data dilakukan dengan cara observasi dan wawancara, serta perancangan sistem menggunakan UML. Metode yang digunakan dalam pengembangan ini adalah scrumban untuk mempermudah mengelola tahapan pengembangan secara sistematis dan adaptif, sehingga dapat menyesuaikan sistem dengan perubahan kebutuhan secara efektif. Dari pengujian blackbox yang telah dilakukan sistem ini mampu memfasilitasi pengajuan secara digital oleh karyawan maupun menyajikan laporan secara otomatis kepada HRD.

**Kata Kunci:** Framework ASP.NET MVC, Scrumban, Sistem Informasi, Website Pengajuan Cuti Izin Lembur

## ABSTRACT

One of the important divisions in Era Medika Hospital is the Human Resource Department (HRD) division which manages human resources including the management of leave, overtime, and employee permits. However, in the process, this submission is still done manually using paper forms which will then be submitted to the relevant superiors. The existing data is then entered into Excel to be used as a report. This risks causing various obstacles such as delays in the approval process, data loss, and lack of efficiency. This study aims to develop a website-based leave, permit, and overtime submission information system to facilitate submission and reporting. This development was carried out using the ASP.NET MVC framework which can optimize the submission and approval process to be more efficient and structured. Data collection was carried out by means of observation and interviews, as well as system design using UML. The method used in this development is scrumban to facilitate managing the development stages systematically and adaptively, so that the system can be adjusted to changing needs effectively. From the blackbox testing that has been carried out, this system is able to facilitate digital submissions by employees and present reports automatically to HRD.

**Keywords:** ASP.NET framework MVC, Information System, Leave Permits Overtime Application Website, Scrumban

## I. PENDAHULUAN

Seiring kemajuan zaman, manusia semakin mengutamakan teknologi dalam kesehariannya. Salah satu penerapannya ialah digunakannya *website* untuk mengganti metode konvensional. *Website* mampu meningkatkan kapasitas serta kecepatan penyampaian informasi untuk penggunanya [1]. Salah satu divisi penting dalam Rumah Sakit Era Medika adalah divisi *Human Resource Departemen* (HRD) yang

memanajemen sumber daya manusia diantaranya adalah pengelolaan cuti, lembur, dan izin karyawan. Cuti adalah kondisi di mana seseorang tidak masuk kerja untuk jangka waktu tertentu dengan izin resmi, yang dapat disebabkan oleh cuti tahunan, persalinan, atau keperluan lainnya. Izin atau ketidakhadiran kerja merupakan kondisi ketika seorang karyawan tidak hadir karena alasan tertentu yang telah ditetapkan dalam kebijakan institusi, instansi, atau perusahaan, serta diatur berdasarkan ketentuan Undang-Undang Ketenagakerjaan Nomor 13 Tahun 2013 [2]. Lembur adalah kondisi dimana karyawan masih bekerja diluar jam kerja dengan tambahan upah tertentu [3]. Pada proses manajemennya pengelolaan cuti, izin, dan lembur karyawan masih dilakukan dengan cara manual yaitu ketika ada karyawan yang ingin mengajukan izin, cuti, atau lembur harus mengisi lembar *form* terlebih dahulu lalu harus memintakan tanda tangan kepada ketua unit, ketua divisi, dan HRD. Jika pihak-pihak tersebut menyetujui, maka karyawan tersebut bisa dinyatakan diizinkan. Hal tersebut akan memperlama pekerjaan dan juga juga beresiko menimbulkan berbagai kendala seperti keterlambatan dalam proses persetujuan, kehilangan data, dan kurangnya efisiensi.

Dalam penyelesaian masalah ini perlu adanya metode pengembangan yang cocok, metode *Scrumban* dipilih untuk pengembangan sistem ini. *Scrumban* adalah metode yang menggabungkan elemen-elemen dari *Scrum* dan *Kanban*. *Scrumban* mengadopsi aspek *Scrum* karena pendekatannya yang terstruktur untuk mencapai kelincahan, serta memanfaatkan prinsip perbaikan berkelanjutan dari *Kanban* untuk memungkinkan tim terus mengoptimalkan proses kerja [4]. *Scrum* memungkinkan pengembangan sistem yang lebih terarah dengan pengelolaan *backlog* dan *sprint* yang teratur [5]. Sementara *Kanban* mendukung visualisasi alur kerja yang efektif, sehingga tim pengembang dapat lebih cepat merespon perubahan kebutuhan.

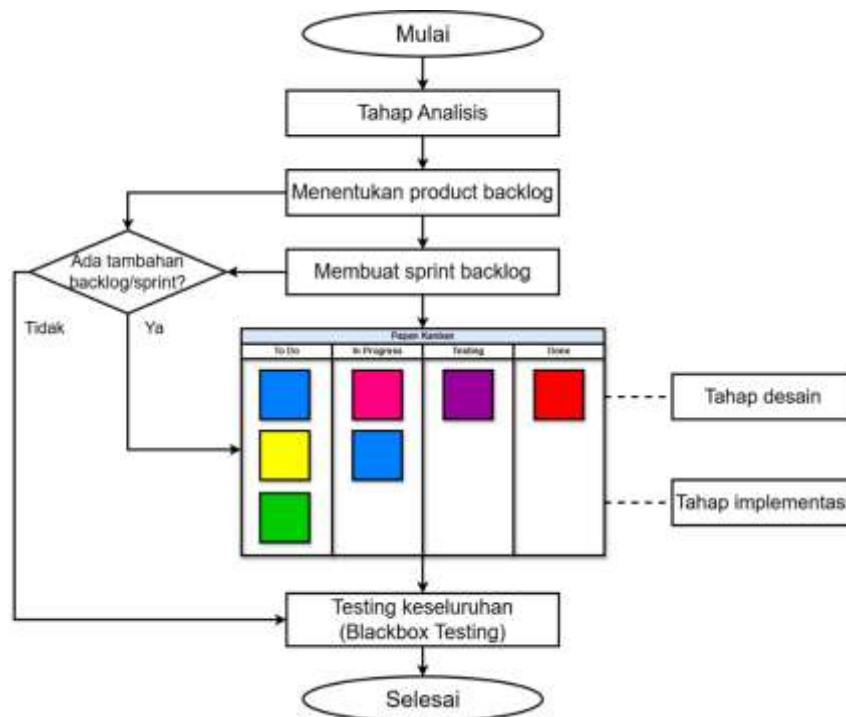
Berdasarkan penelitian terdahulu menyatakan bahwa sistem cuti karyawan yang dibuat berhasil memudahkan karyawan dalam mengakses informasi serta membantu proses pengajuan cuti. Dengan sistem ini, karyawan dapat mengajukan cuti secara lebih mudah tanpa merasa kesulitan dalam menginformasikan maupun mengajukan permohonan cuti [6]. Penelitian lainnya menyatakan pengembangan *website* berhasil dikembangkan dengan menggunakan metode *Scrum* karena penerapan *backlog* didalam *sprint* yang menjadi *guide* dalam pengimplementasian sistem [1]. Penelitian terdahulu lainnya menunjukkan bahwa *framework* ASP.NET sangat cocok dipakai dengan tahap desain yang benar-benar matang mulai dari UML dan *interface* [7]. Penelitian lain yang menggunakan tahap perencanaan observasi, wawancara, dan studi literatur serta analisis kebutuhan secara fungsional dan non-fungsional membuat proses dalam pengembangan sistem menjadi lebih jelas untuk mengetahui fitur apa yang dibutuhkan [8]. Penelitian lainnya menunjukkan bahwa metode *Scrumban*, yang merupakan gabungan antara pendekatan *Scrum* dan *Kanban*, mampu meningkatkan efektivitas tim dalam memenuhi kriteria mutu dimana sudah ditetapkan dan diperkirakan sebelumnya. Selain itu, metode ini juga mendukung efisiensi waktu penyelesaian tugas serta membantu tim dalam mengidentifikasi dan memprioritaskan pekerjaan yang membutuhkan penanganan segera [9]. Sehingga pengembangan sistem pengajuan cuti, izin, dan lembur ini akan efisien menggunakan *framework* ASP.NET dengan pendekatan *Scrumban* karena *Scrum* akan dapat meng-*handle* iterasi yang masuk dan *Kanban* akan membantu memvisualisasikan rencana tahapan yang akan ditempuh.

Implementasi sistem informasi untuk pengajuan cuti, izin, dan lembur diharapkan mampu memberikan kontribusi yang signifikan dalam meningkatkan efisiensi operasional di lingkungan rumah sakit. Dengan sistem ini, proses pengajuan yang sebelumnya manual dan menghabiskan banyak waktu akan diproses secara otomatis, sehingga waktu tunggu persetujuan bisa diminimalisir. Di sisi lain, manajemen rumah sakit Era Medika juga akan lebih terbantu dalam pengelolaan sumber daya manusia, karena data pengajuan yang tersistematisasi.

## II. METODE PENELITIAN

Penelitian ini menggunakan pendekatan *Scrumban* dimana ini adalah kombinasi antara *Scrum* dan *Kanban*. *Scrumban* ini dipilih karena menggabungkan prediktabilitas dan iterasi dari *Scrum* dengan fleksibilitas dan visualisasi alur kerja yang berkelanjutan dari *Kanban*. Akan dibuat beberapa *sprint* berdasarkan *backlog-backlog* yang ada yang merupakan tugas untuk mengerjakan suatu fitur. *Product Backlog* adalah daftar terstruktur yang mencakup semua kebutuhan yang harus dipenuhi dalam sebuah sistem. Daftar ini berfungsi untuk menggali informasi dari tahap sebelumnya serta merinci berbagai

kebutuhan dalam pengembangan sistem informasi [10]. *Sprint* merupakan pecahan pekerjaan, sprint ini akan dimasukkan ke dalam papan *Kanban* dimana akan ada empat kolom yaitu *To Do*, *In Progress*, *Testing*, dan *Done* [11]. Setiap *sprint* memiliki waktu pengerjaan antara 2-4 minggu [12]. Namun pada penelitian ini waktu *sprint* akan dibuat lebih fleksibel sehingga bisa lebih lama ataupun lebih cepat. Keuntungan dari metode ini adalah penggambaran alur kerja yang jelas dan mudah dipahami [13]. Jika terdapat *sprint* yang terhambat akan segera *breakdown backlog* yang ada didalamnya untuk mempermudah pengerjaan dari tugas-tugas yang ada dan menjaga pengelolaan *Work in Progress* (WIP). Namun karena kefleksibilitasnya dalam penerapan *Scrum* harus tetap menjaga prioritas *sprint* untuk dikerjakan terlebih dahulu untuk menjaga keefisienan dalam pengembangan sistem. Pengembangan ini akan dilakukan dengan menggunakan *framework* ASP.NET MVC, *framework* adalah suatu tingkat abstraksi yang diterapkan pada teknologi yang ada, menyediakan struktur dasar untuk pengembangan dan penerapan sistem atau aplikasi [14]. Berikut adalah alur penelitiannya:



**Gambar 1.** Diagram *Work Flow* Penelitian

#### A. Pengumpulan Data

##### 1) Wawancara

Ini merupakan metode interelasi tanpa perantara di antara pengamat serta informan beserta subjek penelitian, yang melibatkan proses tanya jawab. Teknik ini bertujuan untuk memahami lebih dalam terkait pemikiran, pengalaman, dan perspektif dari subjek yang diteliti [15].

##### 2) Observasi

Metode ini adalah suatu prosedur yang dimanfaatkan peneliti untuk memperoleh data dengan melakukan pengamatan langsung pada objek penelitian yang ditetapkan. Pendekatan ini memungkinkan peneliti untuk memperoleh informasi yang akurat dan kontekstual berdasarkan perilaku atau aktivitas yang terjadi di lapangan [15].

##### 3) Studi Pustaka

Studi pustaka mencakup penelaahan terhadap buku, jurnal, serta sumber referensi lain yang relevan dengan pengembangan sistem informasi. Kajian ini meliputi literatur yang membahas perancangan sistem, tahapan pengembangan, dan praktik-praktik terbaik yang digunakan dalam implementasi sistem informasi.

#### B. Desain Sistem

Rancangan sistem dalam penelitian ini disusun melalui tahapan pemodelan menggunakan *Unified*

*Modeling Language* (UML), perancangan antarmuka, dan pengembangan basis data yang mencakup *Conceptual Data Model* (CDM) serta *Physical Data Model* (PDM). Adapun jenis diagram UML yang dimanfaatkan meliputi *use case diagram*, *flowchart*, *activity diagram*, dan *sequence diagram*.

1) *Use Case*

Diagram ini berfungsi untuk memvisualisasikan keterkaitan antara aktor dan sistem, serta mengilustrasikan alur interaksi yang berlangsung dalam menjalankan fungsi-fungsi utama sistem [16].

2) *Flowchart*

*Flowchart* digunakan untuk mempresentasikan tahapan-tahapan dalam suatu proses secara grafis, dengan memanfaatkan simbol-simbol baku dan panah untuk menunjukkan alur informasi atau kendali dari satu tahap ke tahap lainnya [17].

3) *Activity Diagram*

Diagram aktivitas menyajikan alur aktivitas sistem secara grafis, termasuk titik awal, keputusan yang mungkin muncul selama proses berlangsung, hingga akhir dari aktivitas tersebut [16].

4) *Sequence Diagram*

Diagram urutan ini memperlihatkan komunikasi antar objek dalam sistem, termasuk pengguna dan komponen lain, melalui serangkaian pesan yang diatur berdasarkan urutan waktu. Diagram ini memanfaatkan sumbu vertikal sebagai representasi waktu dan sumbu horizontal untuk menunjukkan objek yang terlibat dalam interaksi [18].

5) *Conceptual Data Model (CDM)*

Model ini digunakan untuk mendeskripsikan struktur data secara konseptual, tanpa mempertimbangkan aspek teknis implementasi basis data..

6) *Physical Data Model (PDM)*

PDM menggambarkan struktur basis data secara fisik, termasuk jenis data, struktur tabel, dan elemen teknis lainnya yang dibutuhkan dalam implementasi sistem.

C. *Testing*

Pengujian sistem dilakukan dengan pendekatan *blackbox testing*, yakni teknik menguji suatu perangkat lunak dimana fokusnya di fungsi dan keluaran sistem dengan tidak memfokuskan susunan dalam atau kode sumber program.. Metode ini berfokus pada pengujian fungsionalitas sistem dimana akan dijalankan aplikasi yang akan diuji dan memasukkan banyak jenis data ke dalam setiap formulir, guna mengevaluasi apakah sistem berjalan sebagaimana mestinya secara keseluruhan [19].

### III. HASIL DAN PEMBAHASAN

A. *Data kebutuhan fungsional dan non-fungsional*

Adapun rincian informasi terkait kebutuhan sistem baik fungsional maupun non-fungsional, diperoleh melalui teknik wawancara dan observasi yang sudah dilaksanakan:

1) *Kebutuhan Fungsional*

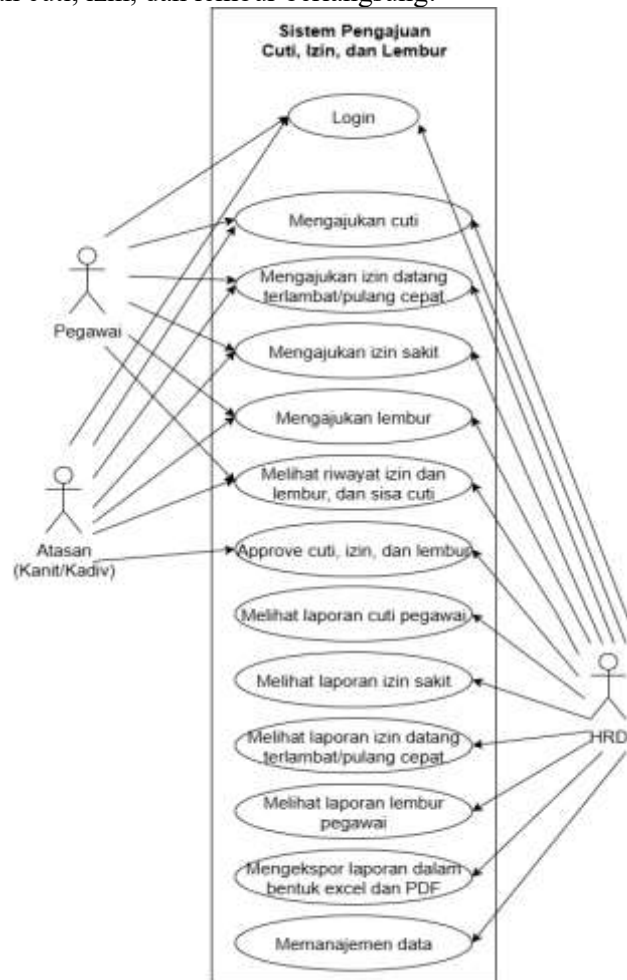
- a. Sistem mampu digunakan untuk mengajukan cuti, izin, dan lembur
- b. Atasan dapat melakukan *approve* pada pengajuan cuti, izin, atau lembur yang dilakukan bawahannya
- c. Sistem dapat menampilkan status cuti, izin, atau lembur yang telah diajukan oleh karyawan
- d. Sistem dapat menampilkan riwayat cuti, izin, atau lembur yang telah diajukan oleh karyawan
- e. Karyawan dapat melihat sisa jatah cuti tahunan yang dimiliki
- f. Sistem dapat menyediakan laporan dari cuti, izin, atau lembur yang telah diajukan oleh karyawan
- g. Sistem dapat mengekspor laporan dari cuti, izin, atau lembur yang telah diajukan oleh karyawan
- h. Sistem dapat digunakan untuk melakukan manajemen data tertentu

2) *Kebutuhan Non-Fungsional*

- a. Desain antarmuka diharapkan intuitif, sederhana, dan memudahkan pengguna dalam bernavigasi, sehingga memberikan kenyamanan dalam penggunaan sistem
- b. Sistem dapat diakses melalui *device* komputer rumah sakit
- c. Sistem harus memiliki autentikasi berupa *username* dan *password* untuk *user*
- d. Sistem dapat dengan baik melakukan pemrosesan berdasarkan aksi yang dilakukan *user*

**B. Use Case**

Tujuan dibuatnya *use case* ini untuk menunjukkan bagaimana aktor berinteraksi dengan sistem selama proses pengajuan cuti, izin, dan lembur:



**Gambar 2. Use Case**

**C. Product Backlog**

Dari hasil kebutuhan fungsional dan non-fungsional yang diuraikan, akan disusun *product backlog*. *Product backlog* yang ke-6 ini adalah tambahan pada proses pengembangan ternyata harus dilakukan pembuatan fitur manajemen data yang sebelumnya tidak ada dalam rencana. Inilah kelebihan *Scrum* sangat fleksibel apabila terdapat penambahan *product backlog* ataupun *sprint*.

**Tabel 1. Product Backlog**

No.	Product Backlog	Prioritas
1	Fitur <i>Login</i>	Tinggi
2	Fitur pengajuan cuti, izin, dan lembur	Tinggi
3	Fitur <i>approval</i>	Tinggi
4	Fitur laporan cuti, izin dan lembur	Tinggi
5	Fitur alur <i>approval</i>	Tinggi
6	Manajemen data (jam kerja pegawai)	Tinggi

**D. Sprint Backlog**

Dalam metode *Scrum* ini, *product backlog* yang sudah ada akan dibuat *sprint backlog*. *Sprint* yang ke-10 ini adalah hasil dari *product backlog* ke-6 yang merupakan tambahan. Berikut adalah *sprint backlog* yang selanjutnya nanti akan berubah menjadi kartu *Kanban*:

**Tabel 2. Sprint Backlog**

<b>Sprint</b>	<b>No.</b>	<b>Kegiatan</b>	<b>Estimasi Waktu</b>
<b>Sprint 1</b>	1	Membuat <i>use case</i>	<b>2 minggu</b>
	2	Membuat <i>flowchart</i>	
	3	Membuat <i>activity diagram</i>	
	4	Membuat <i>sequence diagram</i>	
	5	Membuat rancangan database (CDM, PDM)	
	6	Membuat rancangan <i>interface</i>	
<b>Sprint 2</b>	1	Mempersiapkan <i>tools</i> yang dibutuhkan	<b>2 minggu</b>
	2	Membuat database	
	3	Mengkoneksikan projek dengan database	
<b>Sprint 3</b>	1	Membuat <i>layout</i>	<b>2 minggu</b>
	2	Membuat fitur <i>login</i>	
<b>Sprint 4</b>	1	Membuat fitur <i>form</i> tambah data pengajuan cuti tahunan	<b>3 minggu</b>
	2	Membuat fitur <i>form</i> tambah data pengajuan cuti menikah	
	3	Membuat fitur <i>form</i> tambah data pengajuan cuti melahirkan	
	4	Membuat fitur <i>form</i> tambah data pengajuan izin datang terlambat/pulang cepat	
	5	Membuat fitur <i>form</i> tambah data pengajuan izin sakit	
	6	Membuat fitur <i>form</i> tambah data pengajuan lembur	
<b>Sprint 5</b>	1	Membuat fitur riwayat pengajuan	<b>3 minggu</b>
	2	Membuat fitur alur <i>approval</i>	
<b>Sprint 6</b>	1	Membuat fitur halaman laporan cuti tahunan	<b>3 minggu</b>
	2	Membuat fitur halaman laporan cuti menikah	
	3	Membuat fitur halaman laporan cuti melahirkan	
	4	Membuat fitur halaman laporan rincian izin datang terlambat/pulang cepat	
	5	Membuat fitur halaman laporan izin sakit	
	6	Membuat fitur halaman laporan lembur	
<b>Sprint 7</b>	1	Membuat fitur laporan rekap izin datang terlambat/pulang cepat	<b>2 minggu</b>
	2	Membuat fitur laporan sisa cuti pegawai	
	3	Membuat fitur halaman sisa cuti untuk pegawai	
<b>Sprint 8</b>	1	Membuat fitur ekspor dokumen ke pdf dan excel	<b>1 minggu</b>
	2	Membuat fitur tambahan filter data	
<b>Sprint 9</b>	1	Membuat fitur pengaturan hak akses	<b>2 minggu</b>
	2	<i>Testing Blackbox</i>	
<b>Sprint 10</b>	1	Fitur manajemen data jam kerja	<b>1 minggu</b>

#### E. Implementasi

Pada metode *Scrum* yang diterapkan *tools* yang digunakan untuk memvisualisasikan papan *Kanban* adalah *website trello*. Dalam papan *Kanban* akan dibuat empat kolom yaitu, *To do*, *In progress*, *Testing*, dan *Done*. Dalam pengembangan ini *Work In Progress* (WIP) yang diterapkan sebanyak 2 *sprint* atau dua kartu. Didalam kolom ini akan diisi kartu sebagai visualisasi dari *sprint* yang berjalan. Berikut adalah visualisasi dari papan *Kanban*nya:



Gambar 3. Papan Kanban

1) Halaman Login

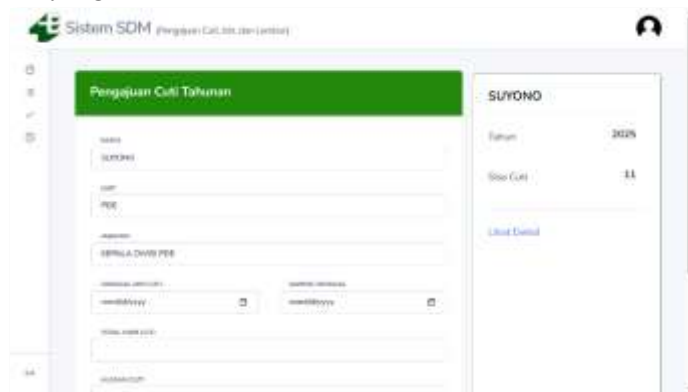
Fitur login ini menjadi pintu utama bagi pengguna, seperti staf, atasan, dan HRD, untuk mengakses sistem berdasarkan peran yang dimilikinya.



Gambar 4. Halaman Login

2) Halaman Pengajuan Cuti Tahunan

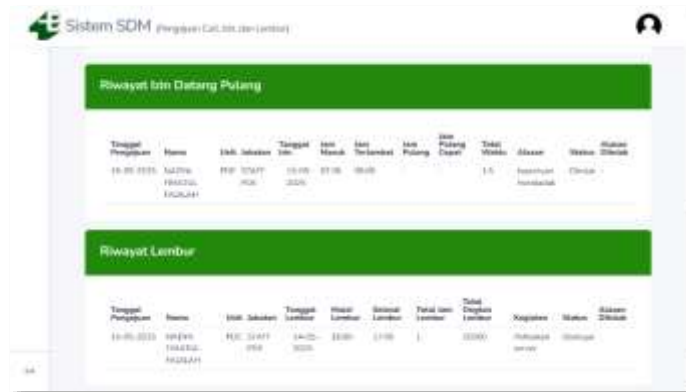
Bagian ini dimanfaatkan pegawai yang akan mengajukan cuti tahunan. Untuk halaman pengajuan lainnya tampilannya akan sama seperti ini, yang membedakan adalah data inputannya yang sudah disesuaikan dengan data yang dibutuhkan.



Gambar 5. Halaman Cuti Tahunan

3) Halaman Riwayat Pengajuan

Halaman ini memungkinkan pegawai untuk mengetahui riwayat pengajuan yang sudah dilakukan.



**Gambar 6.** Halaman Riwayat Pengajuan

4) Halaman *Approval*

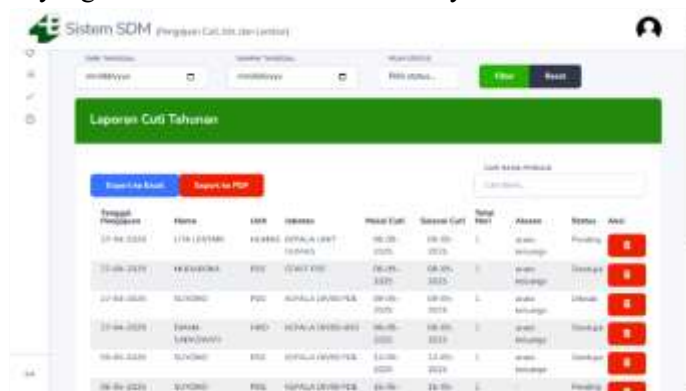
Halaman ini digunakan untuk melihat pengajuan yang dilakukan oleh staf, lalu atasan dan HRD akan menyetujui atau menolak pengajuan yang diajukan.



**Gambar 7.** Halaman *Approval*

5) Halaman Laporan

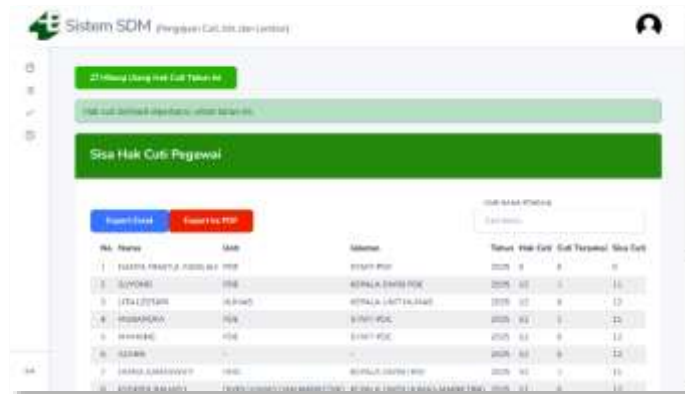
Halaman ini digunakan oleh HRD untuk melihat laporan cuti tahunan. Untuk laporan pengajuan yang lain fiturnya akan sama yang membedakan adalah isi datanya.



**Gambar 8.** Halaman Laporan Cuti Tahunan

6) Halaman Sisa Cuti Pegawai

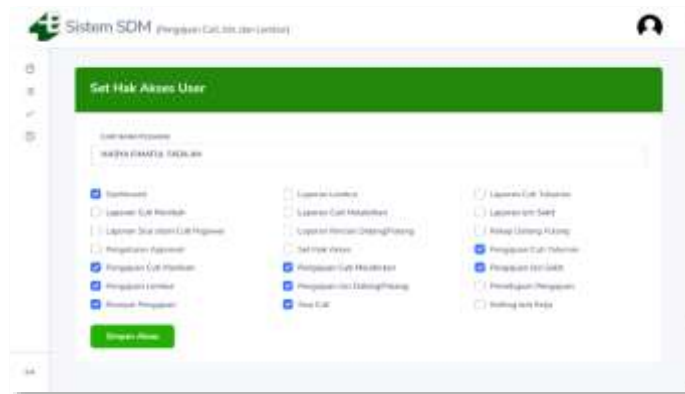
Halaman ini digunakan oleh HRD untuk melihat sisa hak cuti semua pegawai.



Gambar 9. Halaman Laporan Sisa Cuti Pegawai

7) Halaman *Setting Approval*

Halaman ini digunakan untuk mengatur alur *approval* setiap pegawai. Yang bisa mengakses hanyalah HRD.



Gambar 10. Halaman *Setting Approval*

F. *Testing*

Dari pengembangan yang sudah dilakukan, selanjutnya dilakukan pengujian blackbox untuk menguji sistem secara fungsional, berikut adalah pengujiannya:

Tabel 3. Pengujian *Login*

No.	Skenario Uji	Kasus Uji	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1.	<i>Username</i> dan <i>password</i> tidak diisi	<i>Username</i> : (kosong) <i>Password</i> : (kosong)	Sistem menolak akses dan memberikan notifikasi 'Username atau Password salah'	Sesuai dengan hasil yang diharapkan	Valid
2.	<i>Username</i> diisi tetapi <i>password</i> dikosongkan	<i>Username</i> : admin <i>Password</i> : (kosong)	Sistem menolak akses dan memberikan notifikasi 'Username atau Password salah'	Sesuai dengan hasil yang diharapkan	Valid
3.	<i>Username</i> kosong tetapi <i>password</i> diisi	<i>Username</i> : (kosong) <i>Password</i> : admin	Sistem menolak akses dan memberikan notifikasi 'Username atau Password salah'	Sesuai dengan hasil yang diharapkan	Valid

4.	Memasukkan <i>username</i> atau <i>password</i> tidak tepat, klik <i>login</i>	<i>Username</i> : admin12 <i>Password</i> : admin12	Sistem menolak akses dan memberikan notifikasi ‘ <i>Username</i> atau <i>Password</i> salah’	Sesuai dengan hasil yang diharapkan	Valid
5.	Memasukkan <i>username</i> dan <i>password</i> tepat, klik <i>login</i>	<i>Username</i> : admin <i>Password</i> : 12345678	Masuk ke halaman <i>dashboard</i>	Sesuai dengan hasil yang diharapkan	Valid

Berdasarkan Tabel 3 disajikan hasil pengujian blackbox untuk fitur login. Sistem memproses perintah sesuai skenario yang telah ditentukan, dan hasil pengujian cocok dengan apa yang diharapkan. Dari hasil tersebut, disimpulkan bahwa pengujian ini valid dan berhasil.

**Tabel 4.** Pengujian Pengajuan Cuti, Izin, dan Lembur

No.	Skenario Uji	Kasus Uji	Hasil Yang Diharapkan	Hasil Pengajuan	Kesimpulan
1.	<i>Input</i> data dengan ada <i>form</i> yang kosong	Memasukkan data ke dalam <i>form</i> dengan mengosongkan <i>form</i>	Tidak dapat tersubmit dan akan muncul peringatan harus mengisi data yang kosong	Sesuai dengan hasil yang diharapkan	Valid
2.	<i>Input</i> data dengan benar pada <i>form</i> pengajuan	Memasukkan data ke dalam <i>form</i> dengan lengkap dan benar	Dapat tersubmit dan data masuk ke halaman riwayat pengajuan	Sesuai dengan hasil yang diharapkan	Valid

Berdasarkan Tabel 4, pengujian fitur pengajuan cuti, izin, dan lembur menunjukkan bahwa sistem berhasil memvalidasi *input* dengan baik. Pengajuan yang tidak lengkap ditolak dengan pesan peringatan, sedangkan pengajuan yang lengkap berhasil diproses. Maka, pengujian ini dinyatakan valid.

**Tabel 5.** Pengujian Proses *Approval* oleh Atasan

No.	Skenario Pengujian	Test Case	Hasil Yang Diharapkan	Hasil Pengajuan	Kesimpulan
1.	<i>Approve</i> pengajuan	Melakukan klik pada <i>button</i> setuju	Pengajuan sudah tidak muncul lagi	Sesuai dengan hasil yang diharapkan	Valid
2.	Tolak pengajuan	Melakukan klik pada <i>button</i> tolak dan mengisi alasan	Pengajuan sudah tidak muncul lagi	Sesuai dengan hasil yang diharapkan	Valid

Berdasarkan Tabel 5, pengujian terhadap proses *approval* oleh atasan menunjukkan bahwa sistem dapat memproses tindakan setuju maupun tolak dengan benar. Setelah tindakan dilakukan, pengajuan tidak lagi tampil di daftar, hal ini valid dari apa yang yang diinginkan. Dengan demikian, testing ini dinyatakan sesuai.

Berdasarkan Tabel 6, pengujian proses *approval* oleh HRD menunjukkan bahwa sistem mampu memproses keputusan setuju maupun tolak dengan baik. Pengajuan yang telah diproses akan hilang dari daftar dan statusnya berubah sesuai tindakan yang dilakukan. Dengan demikian, pengujian ini dinyatakan valid dan berhasil.

**Tabel 6.** Pengujian Proses *Approval* oleh HRD

No.	Skenario Uji	Kasus Uji	Hasil Yang Diharapkan	Hasil Pengajuan	Kesimpulan
1.	<i>Approve</i> pengajuan	Melakukan klik pada <i>button</i> setuju	Pengajuan sudah tidak muncul lagi dan data sudah berstatus disetujui	Sesuai dengan hasil yang diharapkan	Valid
2.	Tolak pengajuan	Melakukan klik pada <i>button</i> tolak dan mengisi alasan	Pengajuan sudah tidak muncul lagi dan data akan berstatus ditolak	Sesuai dengan hasil yang diharapkan	Valid

**Tabel 7.** Pengujian Manajemen Otomatis Jatah Cuti

No.	Skenario Uji	Kasus Uji	Hasil Yang Diharapkan	Hasil Pengajuan	Kesimpulan
1.	Manajemen otomatis jatah cuti	Mengajukan cuti tahunan	Akan otomatis jatah cuti berkurang yang ditunjukkan di halaman sisa jatah cuti	Sesuai dengan hasil yang diharapkan	Valid
2.	Konversi izin ke cuti	Melakukan pengajuan izin datang terlambat/pulang cepat beberapa kali dengan total jam kerja/hari	Sisa cuti akan otomatis terpotong dengan adanya laporan di rekap izin	Sesuai dengan hasil yang diharapkan jika HRD sudah merekap ulang laporan	Kurang valid

Berdasarkan Tabel 7, fitur manajemen otomatis jatah cuti sebagian besar telah berjalan sesuai harapan. Pengajuan cuti tahunan berhasil mengurangi jatah cuti secara otomatis. Namun, pada skenario konversi izin ke cuti, sistem belum sepenuhnya otomatis karena masih memerlukan rekap manual dari HRD. Maka, pengujian ini menghasilkan satu kasus valid dan satu kasus kurang valid.

**Tabel 8.** Pengujian Manajemen Hak Akses dan Jam Kerja

No.	Skenario Uji	Kasus Uji	Hasil Yang Diharapkan	Hasil Pengajuan	Kesimpulan
1.	Set hak akses <i>user/privilege</i>	Mencari nama <i>user/pegawai</i> , mencentang menu yang dapat diakses, lalu menyimpannya	Data menu yang dapat diakses <i>user</i> dapat tersimpan	Sesuai dengan hasil yang diharapkan	Valid
2.	Mengatur jam kerja pegawai perharinya	Mencari nama pegawai dan memasukkan jam kerja perharinya lalu menyimpannya	Jam kerja pegawai perhari akan tersimpan	Sesuai dengan hasil yang diharapkan	Valid

Berdasarkan Tabel 8, pengujian fitur manajemen hak akses dan pengaturan jam kerja menunjukkan bahwa sistem dapat menyimpan data akses menu dan jam kerja pegawai dengan baik. Semua skenario pengujian memberikan hasil sesuai yang diharapkan, sehingga pengujian ini dinyatakan valid.

G. *Pengujian Keefektifan Scrumban*

Dibawah ini akan disajikan tabel estimasi *sprint* dan waktu implementasinya:

**Tabel 9.** Pengujian Efektivitas *Scrumban*

No.	Sprint	Estimasi Waktu	Waktu Implemetasi	Keterangan
1.	Sprint 1	2 minggu	2 minggu	Tepat
2.	Sprint 2	2 minggu	2 minggu	Tepat
3.	Sprint 3	2 minggu	2 minggu	Tepat
4.	Sprint 4	3 minggu	3 minggu	Tepat
5.	Sprint 5	3 minggu	3 minggu 4 hari	Terlambat
6.	Sprint 6	3 minggu	3 minggu 5 hari	Terlambat
7.	Sprint 7	2 minggu	3 minggu	Terlambat
8.	Sprint 8	1 minggu	1 minggu	Tepat
9.	Sprint 9	2 minggu	2 minggu 2 hari	Terlambat
10.	Sprint 10	1 minggu	1 minggu	Tepat

Berdasarkan Tabel 9 tersebut, diketahui jika total *sprint* yang berhasil diselesaikan tepat waktu sesuai dengan estimasi awal adalah sebanyak 6 *sprint* dari total keseluruhan 10 *sprint* yang direncanakan selama proses pengembangan sistem. *Sprint-sprint* yang dinyatakan tepat waktu tersebut merupakan fase-fase di mana estimasi durasi pengerjaan berhasil dipenuhi sesuai target, tanpa adanya keterlambatan dalam implementasi.

Untuk mengetahui seberapa besar tingkat keberhasilan atau efektivitas penerapan metode *Scrumban* dalam pengelolaan proyek ini, dilakukan perhitungan menggunakan rumus:

$$\text{Keefektifan} = \frac{\text{Jumlah 'Tepat'}}{\text{Total sprint}} \times 100\% \quad (1)$$

$$\text{Keefektifan} = \frac{6}{10} \times 100\%$$

$$\text{Keefektifan} = 60\%$$

Dari penghitungan menggunakan rumus diatas, dapat diambil hasil bahwa keefektifan penerapan *Scrumban* dalam pengembangan ini adalah 60%.

#### IV. KESIMPULAN

Melalui tahapan riset dan pengembangan yang telah dilakukan, diperoleh simpulan bahwa sistem informasi pengajuan cuti, izin, dan lembur yang dikembangkan menggunakan *framework* ASP.NET MVC telah berhasil membuat proses pengajuan dan persetujuan menjadi lebih efisien dan terstruktur karena berhasil menggantikan proses manual yang sebelumnya digunakan oleh Rumah Sakit Era Medika. Sistem ini mampu memfasilitasi pengajuan secara digital oleh karyawan, alur persetujuan oleh atasan dan HRD, serta pelacakan status pengajuan secara *real-time*. Meskipun metode *Scrumban* secara teoritis mendukung fleksibilitas dan visualisasi tugas pengembangan perangkat lunak, implementasinya dalam penelitian ini menghadapi beberapa kendala, terutama dalam hal konsistensi waktu pengerjaan, dibuktikan dengan pengujian keefektifan *Scrumban* yang hanya menghasilkan nilai sebesar 60%. Hal ini dikarenakan keterbatasan waktu yang tersedia selama proses pengembangan, sehingga menyebabkan sejumlah *sprint* tidak dapat diselesaikan sesuai jadwal yang direncanakan. Beberapa *sprint* mengalami keterlambatan sehingga jadwal pengembangan tidak sepenuhnya sesuai dengan perencanaan awal. Meskipun demikian, metode *Scrumban* tetap memberikan kemudahan dalam memantau progres pekerjaan dan menetapkan skala prioritas tugas. Melalui pengujian menggunakan pendekatan *blackbox*, seluruh komponen utama sistem seperti pengajuan, alur persetujuan bertingkat, dan pelaporan, terbukti berfungsi secara optimal sesuai harapan pengguna.

#### DAFTAR PUSTAKA

- [1] N. F. Wibawanto, Y. P. Astuti, N. A. S. Winarsih, G. W. Saraswati, and M. S. Rohman, "Sistem Permohonan Ijin Berbasis Website

- 
- Menggunakan Framework Laravel Dengan Metodologi Scrum,” *J. Manaj. Inform. dan Sist. Inf.*, vol. 6, no. 1, pp. 100–113, 2023.
- [2] F. Fatoni, D. W. Isprananda, and A. Syazili, “Sistem Informasi Pengajuan Cuti dan Izin Berbasis Web,” *J. Sisfokom (Sistem Inf. dan Komputer)*, vol. 9, no. 1, pp. 35–41, 2020, doi: 10.32736/sisfokom.v9i1.712.
- [3] M. Tiardo, “Sistem Perhitungan Upah Lembur Karyawan Pada Kopegtel Lampung,” *J. Ilmu Data*, vol. 2, no. 10, pp. 1–9, 2022, [Online]. Available: <http://ilmudata.org/index.php/ilmudata/article/view/239%0Ahttp://ilmudata.org/index.php/ilmudata/article/download/239/231>
- [4] T. S. Ardan, “Perancangan Sistem Informasi Manajemen Aset Berbasis Web Menggunakan Metode Scrum (Studi Kasus: Universitas Subang),” *J. Tek. Informatika dan Sist. Inf.*, 2024.
- [5] E. D. Wahyuni, N. F. Ramadha, Y. Sangadji, A. Salsabila, and S. Fitriyani, “Analisa Perbandingan Metode Scrum dan Metode Waterfall,” *J. Media Inform. dan Teknol.*, vol. 2, 2024.
- [6] A. M. Ardi, A. Fenty, and L. Lathifah, “Sistem Informasi Pengajuan Cuti Pegawai Menggunakan Metode Pengujian Iso 25010 (Study Kasus : Pt Mutiara Ferindo Internusa),” *J. Inform. dan Rekayasa Perangkat Lunak*, vol. 4, no. 3, pp. 326–334, 2023, doi: 10.33365/jatika.v4i3.3721.
- [7] D. Athallah and D. Hidayatullah, “Penerapan ASP.NET Dalam Membangun System Inventory Dengan Menggunakan Metode Prototype,” *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 9, no. 1, pp. 468–485, 2022, doi: 10.35957/jatisi.v9i1.1559.
- [8] A. Y. Pratama and Y. K. Sari, “Pengembangan Sistem Pengelolaan Non-Akademik Menggunakan Metode Waterfall pada Prodi Informatika Universitas Bhinneka PGRI,” vol. 1, no. 1, pp. 1–9, 2024.
- [9] C. A. Herdian, “Perancangan Question Answering System E-Marketplace Sigertengah Menggunakan Metode Scrum (Studi Kasus : Kelompok Pedagog Sigertengah),” *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 8, no. 2, pp. 547–560, 2021, doi: 10.35957/jatisi.v8i2.830.
- [10] I. Wijayanti, F. E. Nastiti, and A. I. Pradana, “Sistem Informasi Penanganan Komplain Di PT. Tebar Digital Kreasi Dengan Metode Scrum,” vol. 10, no. 2, pp. 252–265, 2023, [Online]. Available: <http://jurnal.mdp.ac.id>
- [11] S. Malakar, *Agile In Practice*, vol. 4, no. 1. New Delhi: BPB Book Center, 2021.
- [12] J. Grotenfelt, “Agile Software Development and Implementation of Scrum,” no. May, 2021, [Online]. Available: [https://www.theseus.fi/bitstream/handle/10024/497598/Agile Software Development and Implementation of Scrum.pdf?sequence=4](https://www.theseus.fi/bitstream/handle/10024/497598/Agile%20Software%20Development%20and%20Implementation%20of%20Scrum.pdf?sequence=4)
- [13] C. Ladas, *Scrum: And Other Essays on Kanban Systems for Lean Software Development*, no. 1. Seattle: Modus Cooperandi Press, 2008. [Online]. Available: [http://scioteca.caf.com/bitstream/handle/123456789/1091/RED2017-Eng-8ene.pdf?sequence=12&isAllowed=y%0Ahttp://dx.doi.org/10.1016/j.regsciurbeco.2008.06.005%0Ahttps://www.researchgate.net/publication/305320484\\_SISTEM\\_PEMBETUNGAN\\_TERPUSAT\\_STRATEGI\\_MELESTARI](http://scioteca.caf.com/bitstream/handle/123456789/1091/RED2017-Eng-8ene.pdf?sequence=12&isAllowed=y%0Ahttp://dx.doi.org/10.1016/j.regsciurbeco.2008.06.005%0Ahttps://www.researchgate.net/publication/305320484_SISTEM_PEMBETUNGAN_TERPUSAT_STRATEGI_MELESTARI)
- [14] F. Soufitri, *Konsep Sistem Informasi*, vol. 3. Medan: PT Inovasi Pratama Internasional, 2023. [Online]. Available: <https://ejournal.upi.edu/index.php/JAPSPs/article/viewFile/6095/4116>
- [15] Z. Iba and A. Wardhana, *Metode Penelitian*. Purbalingga: CV. EUREKA MEDIA AKSARA, 2023.
- [16] F. N. Hasanah and R. S. Untari, *Rekayasa Perangkat Lunak*, 1st ed. UMSIDA Press, 2020. doi: 10.21070/2020/978-623-6833-89-6.
- [17] Y. Khairunisa, E. Mardiani, N. Rahmansyah, A. Ambarwari, Zurati, and A. P. Ariatna, *Dasar-dasar Pemrograman*. PT Penamuda Media, 2023.
- [18] B. Hartono, *Cara Mudah dan Cepat Belajar Pengembangan Sistem Informasi*. Semarang: YAYASAN PRIMA AGUS TEKNIK, 2021.
- [19] T. A. Kurniawan, A. Triayudi, and S. Ningsih, “Implementasi Aplikasi Human Resource Management System (HRMS) Berbasis Web Dan Android Menggunakan Metodologi Agile,” *JURIKOM (Jurnal Ris. Komputer)*, vol. 10, no. 1, p. 212–220, 2023, doi: 10.30865/jurikom.v10i1.5669.