



## Pembangunan Aplikasi Permintaan Cuti Berbasis Web Menggunakan Framework Hibernate dan Spring MVC

Yosef Febrianes<sup>1</sup>, Pratyaksa Ocsa Nugraha Saian<sup>2</sup>

<sup>1,2</sup>Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana

<sup>1</sup>yoseffebrianes@gmail.com, <sup>2</sup>pratyaksa.ocsa@uksw.edu

### Abstract

*PT XYZ is a subsidiary of one of the largest IT companies in Indonesia which is engaged in Information and Communication Technology (ICT). The company's business unit consists of many aspects, for example in the IT distribution business, providing retail for ICT products, outsourcing IT personnel, designing, installing and maintaining all things related to information technology. In order to run all of its businesses, this XYZ company requires a lot of high quality and high quality human resources, but with the outbreak of the Covid19 pandemic since the beginning of 2020, the work process for PT. XYZ is disturbed because it is done in their respective homes. This results in difficulties for employees in processing administrative matters with the office. One of them is in the matter of taking care of leave permits because the previous leave licensing process was still done manually. The management decided to build a special application that can be used by all employees in all subsidiaries. Therefore, the aim of this research is to make an application that can support the administrative needs of employees in the field of leave licensing called "LeaveRequest". This application is designed using the Java programming language with the Spring Boot framework and using the Waterfall software development method. This study found the results that the majority of respondents strongly agree with the design and usability of this application, so it can be concluded that the LeaveRequest application has succeeded in fulfilling the needs in serving the utility of PT. XYZ in work from home condition.*

Keywords: leave, employees, managers, requests, permits

### Abstrak

PT XYZ adalah sebuah anak perusahaan dari salah satu perusahaan IT terbesar di Indonesia yang bergerak dalam bidang Teknologi Informasi dan Komunikasi (TIK). Unit bisnis perusahaan ini terdiri dari banyak aspek, contohnya dalam bidang usaha distribusi IT, menyediakan retail produk-produk TIK, *outsourcing* tenaga IT, perancangan, instalasi, dan perawatan segala hal yang berkaitan dengan teknologi informasi. Demi menjalankan semua bisnisnya, perusahaan XYZ ini membutuhkan banyak sumber daya manusia yang berkualitas dan berkuantitas tinggi, namun dengan merebaknya pandemi Covid19 sejak awal tahun 2020 membuat proses kerja karyawan PT. XYZ ini terganggu karena dilakukan di rumah masing-masing. Hal ini mengakibatkan sulitnya karyawan dalam memproses urusan administrasi dengan kantor. Salah satunya adalah dalam hal mengurus izin cuti karena proses perizinan cuti sebelumnya masih dilakukan secara manual. Pihak manajemen memutuskan untuk melakukan pembangunan sebuah aplikasi khusus yang dapat digunakan oleh semua karyawan pada seluruh anak perusahaan. Oleh karena itu tujuan penelitian ini untuk membuat sebuah aplikasi yang dapat menunjang kebutuhan administrasi karyawan di bidang perizinan cuti yang bernama "LeaveRequest". Aplikasi ini dirancang menggunakan bahasa pemrograman Java dengan *framework* Spring Boot serta menggunakan metode pengembangan perangkat lunak *Waterfall*. Penelitian ini menemukan hasil bahwa mayoritas responden sangat setuju dengan desain dan kegunaan aplikasi ini, sehingga dapat disimpulkan bahwa aplikasi LeaveRequest berhasil memenuhi kebutuhan dalam melayani utilitas permintaan cuti karyawan PT. XYZ pada kondisi work from home.

Kata kunci: cuti, karyawan, manajer, permintaan, perizinan

### 1. Pendahuluan

PT. XYZ adalah sebuah anak perusahaan dari salah satu perusahaan IT terbesar di Indonesia yang bergerak dalam bidang Teknologi Informasi dan Komunikasi

(TIK). Perusahaan ini memiliki banyak unit bisnis, contohnya dalam bidang usaha distribusi dan retail produk-produk TIK, *outsourcing* tenaga kerja dalam bidang IT, menyediakan solusi lengkap mulai dari perancangan desain dan cetak biru, instalasi, konsultasi,



Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.

hingga implementasi, dan perawatan segala hal yang berkaitan dengan teknologi informasi.

Demi menjalankan semua bidang bisnisnya yang luas dan kompleks, perusahaan ini membutuhkan banyak sumber daya manusia yang berkualitas dan berkuantitas tinggi sehingga proses pelayanan terhadap konsumen bisa terlaksana dengan cepat dan baik. Hal ini selaras dengan komitmen pihak manajemen PT. XYZ yaitu berkeinginan untuk memenuhi kepuasan pelanggan dan memberikan layanan terbaik, dengan menerapkan sistem kerja serta operasionalnya agar bisa lebih efektif, terukur, dan terstruktur. Tujuan perusahaan untuk mencapai sistem manajemen mutu agar bisa berjalan dengan baik maka ditetapkan kebijakan serta sasaran mutu ISO/IEC 38500:2016. Kebijakan dan sasaran mutu ini bisa dicapai dengan adanya komitmen dari semua pihak sesuai dengan level jabatannya masing-masing.

Dalam proses mewujudkan komitmennya di tahun 2020, PT. XYZ mengalami situasi yang mengharuskan adanya perombakan sistem kerja yang cukup besar. Hal ini dikarenakan adanya pandemi virus COVID19 yang melumpuhkan semua aspek politik, ekonomi, sosial, budaya, kesejahteraan, serta pertahanan dan keamanan dalam skala dunia[1]. Khususnya di Indonesia, kondisi ini menuntut pemerintah untuk meminimalisir penyebaran virus COVID19 serta menanggulangi dampak yang ditimbulkannya. Demi menjalankan tanggung jawabnya untuk menjaga keberlangsungan dan keselamatan hidup warganya, pemerintah Indonesia tidak henti-hentinya menghimbau kepada seluruh warga negaranya untuk tetap mengikuti anjuran dan mematuhi guna memerangi COVID19, agar penyebaran tidak meluas hingga ke wilayah lainnya di Indonesia.

Salah satu upaya pemerintah dalam memerangi pandemi COVID19 adalah dengan dijalkannya Undang-Undang Nomor 6 Tahun 2018 Tentang Kesehatan, khususnya dijelaskan dalam pasal 1 ayat 10 yang berbunyi: “Karantina Wilayah adalah pembatasan penduduk dalam suatu wilayah termasuk wilayah pintu masuk beserta isinya yang diduga terinfeksi penyakit dan/atau terkontaminasi sedemikian rupa untuk mencegah kemungkinan penyebaran penyakit atau kontaminasi”[1][2]. Hal inilah yang menuntut PT. XYZ untuk memberlakukan pembatasan aktivitas di dalam kantor sehingga proses bekerja dilakukan di rumah atau work from home (WFH). Demi tetap menunjang kegiatan bekerja para karyawannya agar sesuai dengan standar mutu ISO/IEC 38500:2016, maka dibuatkanlah aplikasi-aplikasi berbasis web sehingga karyawan dapat tetap bekerja dengan efektif. Dari sekian banyak aplikasi penunjang, salah satunya adalah aplikasi LeaveRequest yang berfungsi untuk menunjang permintaan cuti karyawan terhadap atasannya sehingga hak karyawan dalam mengambil hari cuti dapat terpenuhi dengan baik secara online[3]. Aplikasi LeaveRequest adalah sebuah aplikasi pengajuan cuti karyawan berbasis web yang

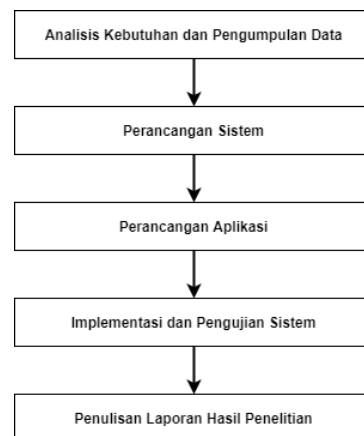
dapat mengolah segala data cuti karyawan sehingga segala proses pengajuan dan perizinan cuti di PT. XYZ dapat berjalan lancar, aman, dan memiliki sistem pendataan yang jelas.

Cuti merupakan hak yang diterima karyawan untuk tidak masuk kerja dalam jangka waktu tertentu untuk menjamin kesegaran rohani dan jasmani karyawan yang bekerja dalam perusahaannya[4][5]. Menurut Peraturan Pemerintah Nomor 24 Tahun 1976, Cuti adalah keadaan tidak masuk kerja yang diijinkan dalam jangka waktu tertentu. Jenis-jenis cuti tersebut terdiri dari cuti tahunan, cuti besar, cuti sakit, cuti bersalin dan sebagainya[6].

Sebelumnya proses pengajuan cuti di PT. XYZ dinilai cukup rumit dan panjang. Hal ini semakin diperparah dengan adanya pandemi COVID19 yang membuat adanya jarak antara karyawan dengan atasannya sehingga proses perizinan semakin sulit dan lambat. Oleh karena itu dibutuhkanlah sebuah sistem informasi yang memadai untuk memangkas proses yang kurang efektif pada sistem yang ada sebelumnya. Sistem informasi yang baik adalah dihasilkan dari hubungan kerja sama dalam mengumpulkan, memproses, menyimpan dan menyebarkan informasi sebagai dasar pengambilan suatu keputusan[7][8]. Dilihat berdasarkan permasalahan yang ada di lapangan maka penelitian ini dibuat untuk membuat sebuah aplikasi demi menunjang permintaan dan perizinan cuti karyawan yang bernama LeaveRequest.

## 2. Metode Penelitian

Perancangan aplikasi ini dilakukan dengan melewati 5 tahapan yaitu: (1) analisis kebutuhan dan pengumpulan data; (2) perancangan sistem; (3) perancangan aplikasi; (4) implementasi dan pengujian sistem dengan cara simulasi; dan (5) penulisan laporan hasil penelitian. Alur tahapan penelitian pada penelitian ini dapat dilihat pada Gambar 1 dibawah ini.

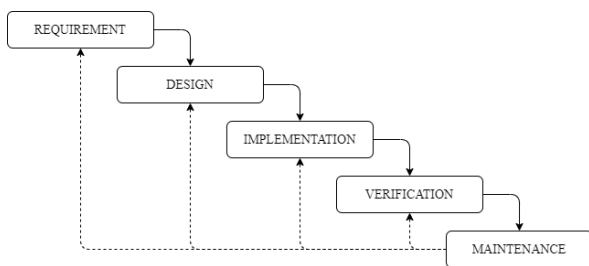


Gambar 1. Tahapan Penelitian

Berdasarkan pada Gambar 1, dapat dijelaskan bahwa tahap pertama atau tahap analisis kebutuhan dan pengumpulan data merupakan proses dimana

pengembang menganalisis serta mempersiapkan *tools* yang akan dibutuhkan selama membangun aplikasi LeaveRequest ini. *Tools* yang disepakati ini adalah penggunaan *dependency* DevTools, Spring Web, Lombok, Spring Data JPA, dan lain-lain. Selanjutnya proses hosting dilakukan menggunakan Apache Tomcat Server. Selain mencari sumber referensi dari penelitian terdahulu, pada tahap ini pihak pengembang juga mencari semua informasi mengenai kebutuhan dari pihak pengguna yang berhubungan proses pengembangan sistem. Proses pengumpulan informasi dilakukan dengan cara berdiskusi dengan beberapa *staff* IT yang bekerja di PT. XYZ. Berdasarkan hasil diskusi tersebut, disimpulkan bahwa PT. XYZ memang memerlukan sistem manajemen untuk mengelola hal-hal yang berkaitan dengan cuti karyawan. Terlebih setelah ditimbang pada situasi pandemi global ini yang membuat pemerintah menerbitkan aturan bahwa setiap kantor hanya boleh beroperasi dengan kapasitas 25% dari total kapasitasnya, hal ini mendorong dibutuhkan sistem yang terintegrasi secara daring dalam mengelola segala hal yang berkaitan dengan proses administrasi karyawan, khususnya dalam hal permintaan dan perizinan cuti. Didapatkan juga info bahwa selama ini PT. XYZ belum memiliki sistem manajemen pengelolaan cuti, sehingga proses pengajuan dan perizinan cuti masih berjalan secara manual. Tentunya hal ini memiliki banyak sekali celah untuk ada terjadinya kesalahan dalam proses baik disengaja maupun tidak disengaja (*human error*).

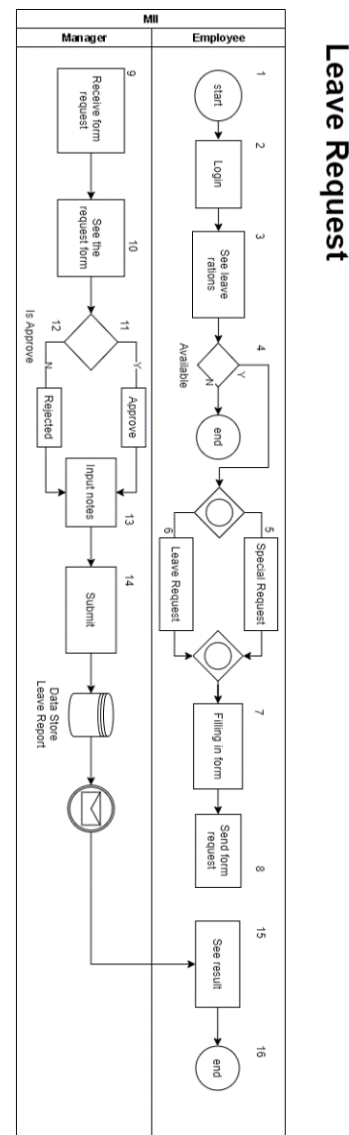
Proses 2 dan proses 3 yang merupakan tahap perancangan sistem dan tahap pembuatan aplikasi pada penelitian ini menggunakan metode *Waterfall*. Metode pengembangan *Waterfall* adalah salah satu metode yang paling banyak digunakan dalam proses pembuatan aplikasi. Alur dari metode *Waterfall* yang berjalan secara simpel dan alami membuat metode ini cocok digunakan oleh pemula. Tak hanya itu, prosesnya yang tidak memiliki banyak aturan juga memberi kemudahan kepada pengembang karena tidak banyak membatasi proses.



Gambar 2. Metode *Waterfall*

Proses pada tahap perancangan aplikasi menghasilkan beberapa *output*, diantaranya berupa *entity relationship diagram* (ERD), *business process model and notation* (BPMN), dan *use case diagram* (UCD). ERD adalah sebuah diagram yang berfungsi untuk menggambarkan hubungan antara tabel-tabel yang ada pada sebuah basis

data. Hubungan antara tabel-tabel tersebut direlasikan melalui entitas-entitas yang terdapat dari masing-masing tabel tersebut. BPMN adalah sebuah penggambaran model proses bisnis yang di dalamnya terdapat notasi untuk menggambarkan alur-alur yang dilalui pengguna saat menggunakan aplikasi tersebut. BPMN juga merupakan standar dalam penggambaran alur kerja sebuah aplikasi yang mirip dengan *unified modeling language* (UML). Diharapkan dengan adanya simbol-simbol yang mudah dipahami pada BPMN, pengguna khususnya orang awam dari bidang IT bisa untuk lebih mudah dalam memahami alur proses dalam sebuah sistem. UCD adalah sebuah diagram yang berfungsi untuk menggambarkan hubungan interaksi antara aktor dengan sebuah sistem yang bekerja. Dalam UCD juga dapat diketahui siapa saja aktor yang berada dalam sebuah sistem. User juga bisa mengetahui apa saja fitur-fitur yang dapat digunakan oleh aktor-aktor tersebut.



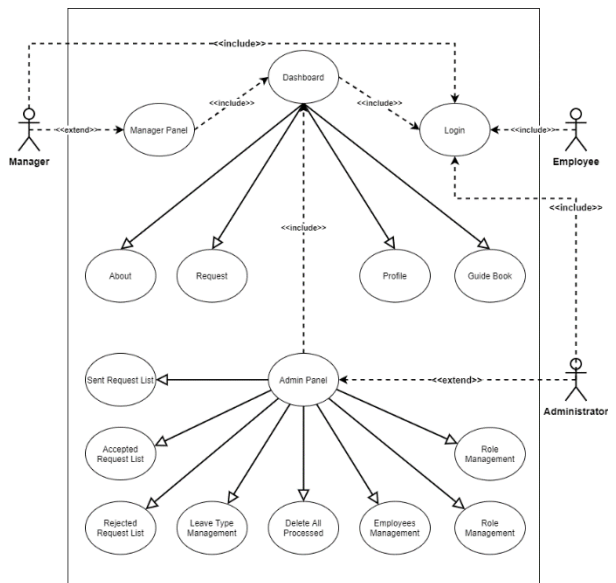
Gambar 3. *Business Process Model and Notation*

Gambar 3 merupakan BPMN dari aplikasi LeaveRequest. Diagram tersebut menggambarkan alur

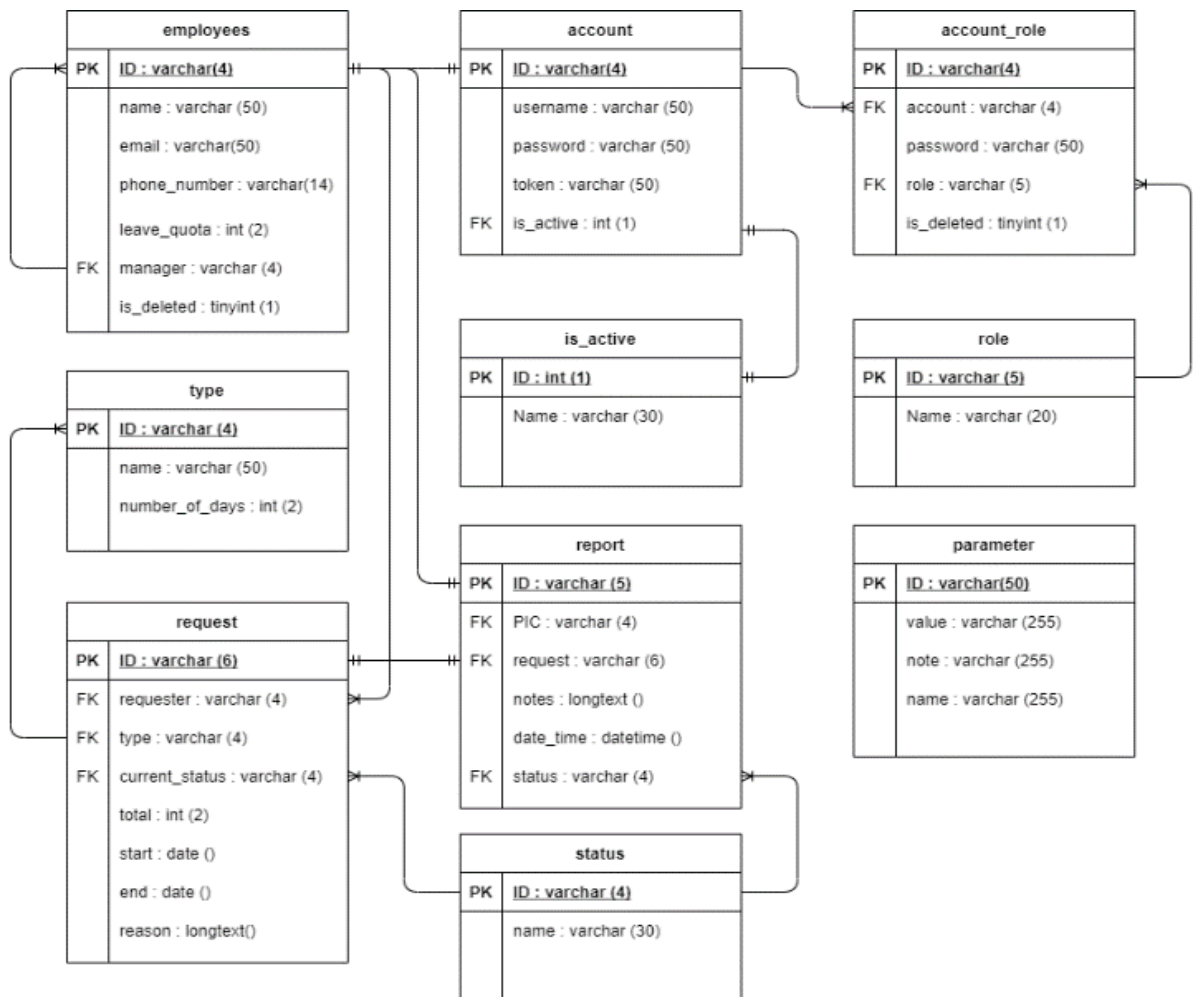
proses bisnis yang terjadi antara *role employee* dan *role manager*. Prosesnya dimulai dari *employee* login ke dalam aplikasi, lalu melihat jumlah kuota cuti yang mereka miliki. Jika masih memiliki kuota cuti yang tersisa, maka *employee* bisa memasuki tahap selanjutnya yaitu mengisi formulir permintaan cuti (proses 6), dan mengirimkan formulirnya. Sistem akan otomatis mengirimkan formulir tersebut menuju *manager* dari *employee* tersebut. Setelah formulir selesai terkirim, maka *manager* dari *employee* tersebut akan mendapat pemberitahuan (proses 9). Langkah selanjutnya adalah *manager* memberi pertimbangan dan menentukan apakah permintaan cuti dari *employee* tersebut akan disetujui atau tidak. Apapun keputusan yang diambil *manager*, sistem akan mengirimkan pemberitahuan hasilnya kepada *employee* berupa email yang otomatis terkirim menuju email milik *employee* tersebut yang terdaftar. Setelah email sampai, maka *employee* tersebut sudah dapat melihat hasil perizinannya (proses 15).

Gambar 4 merupakan ERD dari basis data sistem LeaveRequest. Basis data tersebut berjumlah 10 tabel yang terdiri dari 2 bagian utama, yaitu user credential dan business management. User credential berfungsi untuk menyimpan dan mengolah setiap informasi yang

berkaitan dengan data pengguna. Sedangkan business management merupakan bagian yang berguna untuk menyimpan dan mengolah transaksi bisnis pada sistem, dalam hal ini tentang data permintaan dan perizinan cuti.



Gambar 5. Use Case Diagram LeaveRequest.



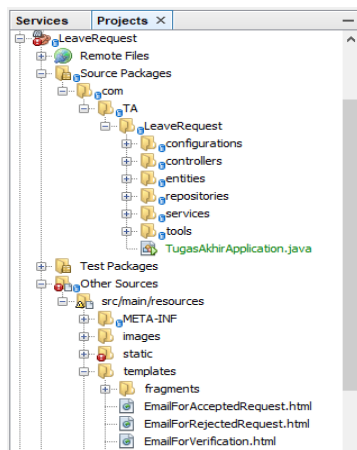
Gambar 4. Entity Relationship Diagram

Hasil terakhir dari perancangan sistem dalam *unified modeling language* adalah *use case diagram* yang ada pada Gambar 5. *Use case diagram* sistem LeaveRequest terdiri dari tiga aktor yaitu *employee*, *manager*, dan *administrator*. Semua aktor yang ada disini diwajibkan untuk *login* ke dalam aplikasi terlebih dahulu. Setelah pengguna *login*, maka akan langsung terarah menuju halaman dasbor. Aktor *employee* disini hanya bisa mengakses panel *about*, *request*, *profile*, dan *guide book*. Aktor *manager* bisa mengakses *manager panel* dan semua halaman yang dapat diakses oleh aktor *employee*. Sedangkan aktor *administrator* bisa mengakses semua halaman yang ada pada aplikasi LeaveRequest.

Dari Gambar 5 bisa dilihat secara garis besar bahwa aplikasi LeaveRequest ini terbagi menjadi 3 panel utama, yaitu panel *dashboard* yang bisa diakses semua aktor, panel *manager* yang bisa diakses oleh *manager* dan *administrator* dan terakhir panel *administrator* yang hanya bisa diakses oleh pengguna yang memiliki *role administrator*. Panel *dashboard* berfungsi untuk menjadi jembatan terhadap halaman-halaman yang lain yang ada pada aplikasi LeaveRequest. Panel *manager* adalah halaman dimana manajer melihat semua data karyawan yang mengirimkan permintaan cuti. Pada halaman ini juga seorang manajer menolak atau mengizinkan permintaan cuti dari seorang karyawan. Terakhir merupakan panel *administrator* dimana seorang admin bisa melihat dan mengelola semua data yang berkaitan dengan *user management* dan *business process* dari aplikasi LeaveRequest ini.

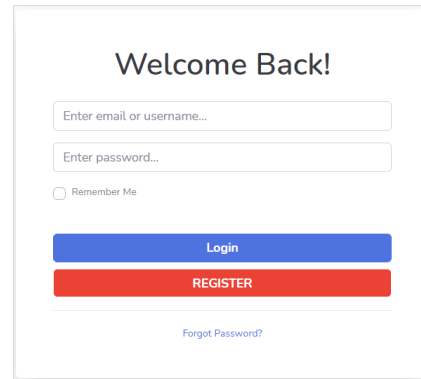
### 3. Hasil dan Pembahasan

Hasil implementasi dari kerangka kerja MVC dapat dilihat di Gambar 6 dimana struktur pengkodean aplikasi LeaveRequest dibagi menjadi beberapa *Java package* untuk memisahkan jenis-jenis kelas sesuai dengan fungsinya. Hal ini bertujuan untuk menghindari proses *maintenance* yang lama dikarenakan pengguna pada aplikasi ini merupakan *thick client* dimana *client* ini akan juga menangani *business logic process*[9]. Struktur MVC disini mengalami penyesuaian dengan adanya kolaborasi *framework* lain yaitu Spring MVC.



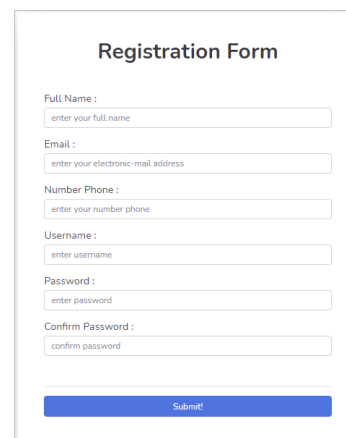
Gambar 6. Struktur kode LeaveRequest

*Package* utama dalam implementasi Spring MVC disini adalah adanya *entities/model*, *service*, dan *controller*[9]. Sedangkan pada *package configurations* dan *tools* berguna untuk menyimpan segala konfigurasi dan kelas-kelas lain yang digunakan pada proses pengembangan aplikasi LeaveRequest ini seperti Bcrypt, konfigurasi RBAC, kelas untuk pembentukan manipulasi String saat meng-*generate* nomor identitas setiap data yang disimpan di dalam basis data.



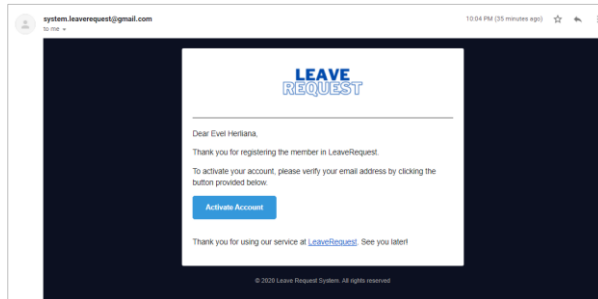
Gambar 7. Tampilan Form Login

Gambar 7 adalah tampilan dari formulir login. Pada halaman ini, pengguna bisa login menggunakan nama pengguna atau alamat email yang sudah terdaftar saat melakukan registrasi. Pengguna yang belum memiliki akun maka bisa mengklik tombol register sehingga akan langsung dialihkan ke halaman registrasi akun. Sedangkan pada gambar 8 merupakan tampilan dari formulir registrasi akun. Pada bagian ini, pengguna diwajibkan untuk mengisi data-data yang diminta yaitu nama lengkap, alamat email, nomor telepon, nama pengguna, dan kata sandi. Setelah pengguna melengkapi formulir registrasi dan mengklik tombol submit, maka sistem akan secara otomatis mengirimkan pesan aktivasi akun ke alamat email yang didaftarkan. Pada tahap ini akun yang berhasil didaftarkan belum bisa digunakan karena akun tersebut masih belum aktif, sehingga perlu proses pengaktifan akun.



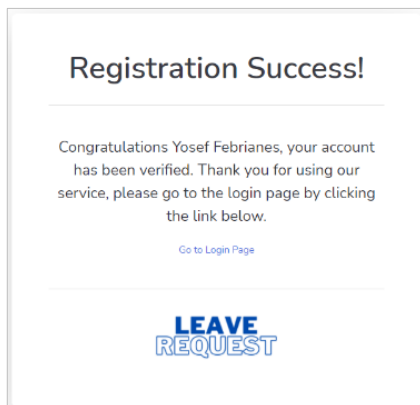
Gambar 8. Tampilan Form Registrasi

Pesan pada email tersebut bisa dilihat di gambar 9, dimana terdapat pesan pemberitahuan bahwa pengguna telah selesai mendaftarkan akun miliknya di sistem aplikasi Leave Request. Pada tahap ini, pengguna diminta untuk menekan tombol “Activate Account” yang berwarna biru untuk mengaktifkan akun yang telah didaftarkan.



Gambar 9. Template Verifikasi Akun pada Email

Setelah tombol “Activate Account” ditekan, maka pengguna akan dialihkan ke sebuah halaman yang berisi pesan bahwa akun yang didaftarkan telah berhasil diaktifkan. Pemberitahuan tersebut bisa dilihat pada gambar 10.

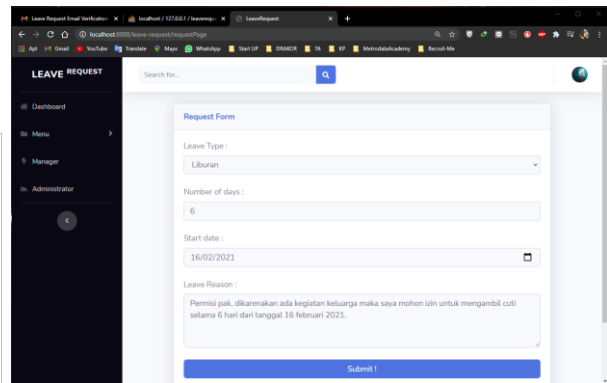


Gambar 10. Pesan Proses Registrasi Selesai

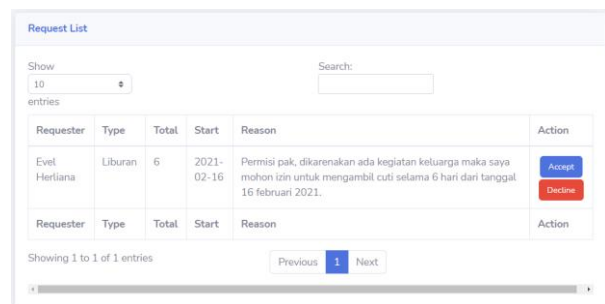
Setelah pengguna telah mendaftarkan akunnya, maka pengguna sudah bisa menggunakan akun tersebut untuk melakukan transaksi permintaan cuti. Sebelum memasuki halaman permintaan cuti, pengguna bisa melihat jumlah kuota cuti yang mereka miliki pada halaman *profile* pada *dashboard*. Setelah mengetahui info *profile* serta jumlah cuti yang dimiliki, pengguna sudah bisa melakukan transaksi perizinan permintaan cuti sesuai dengan jumlah kuota cuti yang mereka miliki. Gambar 11 berikut merupakan formulir pendaftaran permintaan cuti pada sistem LeaveRequest.

Setelah pengguna mengirimkan formulir permintaan izin, maka secara otomatis manager dari pengguna tersebut akan mendapatkan notifikasi. Proses selanjutnya adalah dimana sang manajer membuat keputusan terhadap permintaan cuti tersebut. Halaman pengambilan keputusan tersebut bisa dilihat pada

gambar 12. Pada tabel terdapat beberapa data seputar permintaan cuti dari karyawan seperti nama, tipe cuti, jumlah hari, tanggal, dan alasan cuti.

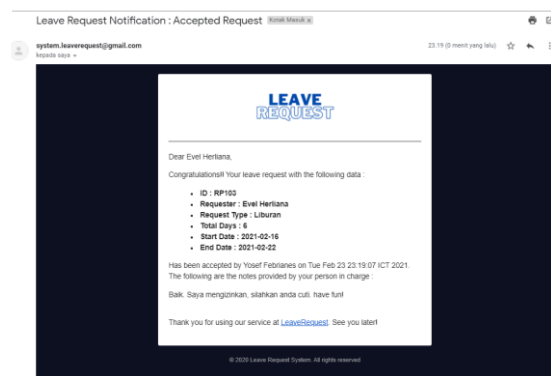


Gambar 11. Formulir Permintaan Cuti



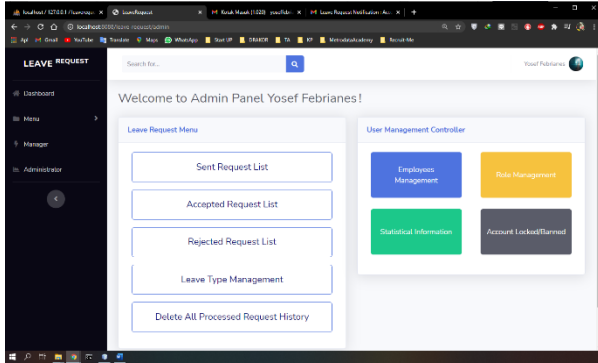
Gambar 12. Tampilan Manager Panel

Setelah izin dikirimkan, sistem akan otomatis mengirimkan pemberitahuan kepada karyawan melalui email. Gambar 14 merupakan tampilan pesan pada email karyawan saat permintaan cuti diterima. Pesan tersebut berisi ucapan selamat dan pemberitahuan bahwa permintaan cuti mereka diterima. Selain itu ditampilkan informasi yang berkaitan dengan data cuti, serta pesan dari *Person In Charge* (PIC) atau manager sang karyawan itu sendiri.



Gambar 14. Tampilan Pesan Saat Cuti Diterima Pada Email

Pada panel admin terdapat banyak fitur yang tersedia dalam menu di *dashboard administration*. Menu tersebut dibagi menjadi 2 yaitu *Leave Request Menu* dan *User management Controller* untuk membedakan dalam mengelola data tentang cuti dengan data tentang karyawan.



Gambar 15. Tampilan *Dashboard Administration*

Function `sendVerificationEmail()`

```

1 public String
2 sendVerificationEmail(Employees
3 employees, Context context)
4     throws MessagingException {
5
6     String response;
7     try {
8         MimeMessage message =
9 sender.createMimeMessage();
10        MimeMessageHelper helper =
11 new MimeMessageHelper(message,
12 MimeMessageHelper.MULTIPART_MODE_MIXE
13 D_RELATED,
14 StandardCharsets.UTF_8.name());
15
16        String html =
17 templateEngine.process("EmailForVerif
18 ication", context);
19        helper.setText(html, true);
20        helper.setFrom(email);
21
22        helper.setTo(employees.getEmail());
23        helper.setText(html, true);
24        helper.setSubject("Leave
25 Request Email Verification: ");
26        sender.send(message);
27        response = "true";
28
29    } catch (MessagingException e) {
30        response = ("Mail sending
31 failure : " + e.toString());
32    }
33    return response;
34 }
35 }
36 }
    
```

Kode program 1 menunjukkan cara mengirimkan email untuk verifikasi akun dengan cara mengambil data dari basis data atau yang lebih spesifik dari tabel employee. Setelah mengetahui alamat email yang akan dituju, maka sintaks `helper.setTo` diset kepada alamat email milik pengguna (baris 24). Kode `sendVerificationEmail` merupakan salah satu dari beberapa fungsi pada kelas `EmailService` dimana ini merupakan kelas dari penerapan teknologi Java Mail Sender.

Kode program 2 berfungsi untuk melakukan pengaturan atau konfigurasi pada Role-Based Action Control. RBAC berfungsi untuk meningkatkan keamanan sebuah sistem dimana bekerja untuk membatasi hak akses setiap penggunanya [10]. Dalam aplikasi `LeaveRequest`, setiap pengguna dibatasi sesuai peran yang dimiliki sang penggunanya. Hasil implementasi RBAC disini adalah

dimana seorang dengan peran `employee` tidak dapat memasuki halaman untuk pengguna yang memiliki peran `administrator`.

Procedure `configure()`

```

1 @Configuration
2 @EnableWebSecurity
3 @EnableGlobalMethodSecurity(securedEnab
4 led = true, proxyTargetClass = true)
5 public class SecurityConfiguration {
6     extends WebSecurityConfigurerAdapter {
7
8     @Override
9     protected void configure(HttpSecurity
10 http) throws Exception {
11         http.authorizeRequests()
12
13             .antMatchers("/admin").hasRole("ADMIN")
14
15             .antMatchers("/dashboardAdministrator")
16             .hasRole("ADMIN")
17
18             .antMatchers("/dashboardManager").hasRo
19 le("MANAGER")
20
21             .antMatchers("/dashboard").hasAnyRole("
22 EMPLOYEE", "MANAGER", "ADMIN")
23
24             .antMatchers("/manager").hasAnyRole("AD
25 MIN", "MANAGER")
26
27             .antMatchers("/requestPage").hasAnyRole
28 ("ADMIN", "MANAGER", "EMPLOYEE")
29             .antMatchers("/").permitAll()
30             .and()
31             .formLogin()
32             .loginPage("/login")
33             .defaultSuccessUrl("/")
34             .failureUrl("/loginerror")
35             .permitAll()
36             .and()
37             .logout()
38             .logoutRequestMatcher(new
39 AntPathRequestMatcher("/logout"))
40
41             .logoutSuccessUrl("/logoutsuccess")
42             .deleteCookies("my-remember-me-
43 cookie")
44             .permitAll()
45             .and()
46             .rememberMe()
47             .rememberMeCookieName("-my-
48 remember-me-cookie")
49             .and()
50             .exceptionHandling()
51             .and()
52
53             .exceptionHandling().accessDeniedPage("
54 /accessDenied");
55     }
56 }
    
```

Class `Generate()`

```

1 public class Generate {
2
3     public static String id(String
4 prefix, long size, int length) {
5         String temp = prefix;
6         String count = (size + 1) + "";
7         for (int i = prefix.length() +
8 count.length(); i < length; i++) {
9             temp += 0;
10        }
11        temp += count;
12        return temp;
13    }
14 }
    
```

Kode program 3 diambil dari kelas `Generate`. Kelas

generate ini menyimpan fungsi untuk memanipulasi data dengan tipe String untuk dijadikan kode sebagai *identifier* sebuah data saat disimpan di dalam basis data. Fungsi dengan nama id ini berfungsi untuk *generate* sebuah variabel sehingga berbentuk seperti sesuai yang diinginkan. Cara kerja dari fungsi ini adalah dengan memasukkan tiga variabel diantaranya variabel prefix dengan tipe data String, variabel size dengan tipe data long, dan tipe data integer. Prefix dimasukan sebuah alfabet yang diinginkan, biasanya kode alfabet tersebut memiliki makna dari sebuah data tersebut. Variabel size adalah jumlah seluruh data yang ada pada tabel sebuah basis data tersebut. Pengembang menggunakan fungsi `.count()` untuk mendapatkan jumlah total data pada tabel tersebut. Dan terakhir variabel `length` adalah jumlah karakter yang diinginkan pada *identifier* tersebut.

Pengujian sistem pada penelitian ini menggunakan 2 cara, yaitu metode pengujian *black-box testing* dan metode *usability testing*. *Black-box testing* merupakan sebuah metode pengujian sistem yang berfungsi untuk menguji dan meneliti fungsionalitas sebuah perangkat lunak tanpa harus melihat struktur yang ada di dalam aplikasi, dalam hal ini *source code* dan basis data dari aplikasi tersebut. Pengujian *black-box testing* ini dilakukan agar dapat disimpulkan bahwa aplikasi LeaveRequest ini dapat berjalan dengan baik tanpa ada bug dan bisa berjalan sesuai dengan yang diharapkan. Proses pengujian ini adalah dengan membandingkan antara hasil yang diharapkan oleh pengembang dengan hasil dari pengujian terhadap fungsi-fungsi yang ada dalam sistem oleh sang penguji[11].

Sedangkan pengujian dengan metode *usability testing* digunakan untuk mengevaluasi kualitas aplikasi LeaveRequest ini, khususnya dalam segi *user usability*, *user interface* dan *user experience*. Pengujian dengan menggunakan sudut pandang *user*(pengguna) di luar dari tim developer (pengembang aplikasi) sangatlah berguna karena *user* awam dapat melihat sebuah aplikasi dengan sudut pandang yang berbeda sehingga itulah mengapa metode ini sangat penting untuk dilakukan[11].

Tabel 1 mencatat hasil dari *black-box testing*. Proses pengujian ini dilakukan oleh seorang *Quality Assurance* dan 2 orang *Application Developer Junior* selama 2 hari. Setelah melalui proses pengujian maka didapatkan hasil dimana setiap variabel pengujian mendapatkan status yang *valid*. Berdasarkan hasil pengujian sistem menggunakan *black-box testing*, dapat disimpulkan bahwa sistem LeaveRequest berjalan dengan baik dan sesuai dengan yang diharapkan.

Tabel 1. Hasil *Black-Box Testing*

No	Deskripsi	Hasil yang Diharapkan	Hasil Pengujian
1	Mengirim email setelah melakukan registrasi akun.	Sistem secara otomatis mengirimkan pesan verifikasi akun ke alamat email pengguna.	Berjalan baik
2	Sistem pengamanan akun terhadap <i>brute force attack</i> .	Akun otomatis terkunci setelah 3 kali salah memasukkan <i>password</i> .	Berjalan baik
3	<i>Login</i> menggunakan <i>email</i> atau <i>username</i> dan <i>password</i> .	Pengguna dengan akun yang telah terverifikasi dapat masuk ke aplikasi.	Berjalan baik
4	Pengujian pada sistem RBAC.	Hak akses setiap pengguna dibatasi sesuai dengan <i>Role</i> -nya masing-masing.	Berjalan baik
5	Pengujian terhadap proses pengiriman <i>request</i> cuti pada panel <i>employee</i> .	Pengguna yang memiliki cukup kuota cuti dapat melakukan proses <i>request</i> , semua data <i>request</i> tersimpan di <i>database</i> .	Berjalan baik
6	Pengujian proses perizinan cuti pada panel <i>Manager</i> .	Proses <i>accept</i> , <i>decline</i> , <i>read</i> , dan <i>insert</i> data berjalan baik. Pengiriman pemberitahuan otomatis oleh sistem tentang keputusan <i>manager</i> kepada email <i>requester</i> .	Berjalan baik
7	Pengujian data <i>history</i> pada semua transaksi.	Data tersimpan dengan baik di <i>database</i> dan dapat ditampilkan pada tabel <i>history</i> .	Berjalan baik
8	Pengujian fungsi <i>create</i> , <i>read</i> , <i>update</i> , dan <i>delete</i> di semua panel.	Fungsi CRUD berjalan dengan baik tanpa ada <i>data crash</i> .	Berjalan baik

Tabel 2. Daftar Pertanyaan pada metode *Usability Testing* Kategori Kegunaan Sistem (*System Usability*)

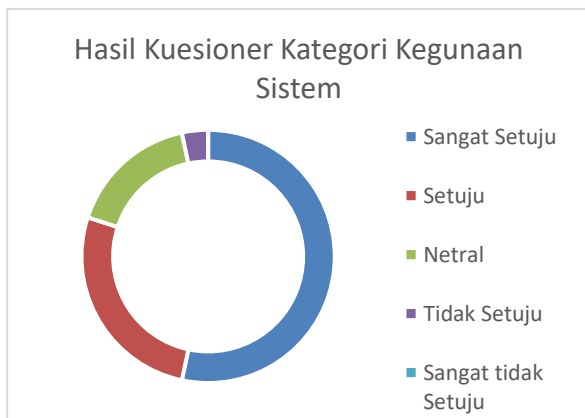
No	Daftar Pertanyaan Skala (Kategori Kegunaan Sistem)	Penilaian
1	Apakah aplikasi mudah digunakan?	1 2 3 4 5
2	Apakah informasi dalam aplikasi mudah untuk diakses?	1 2 3 4 5
3	Apakah proses <i>loading data</i> pada saat penggunaan aplikasi berjalan dengan baik, cepat dan lancar?	1 2 3 4 5
4	Apakah data yang diolah menjadi informasi mudah untuk dipahami?	1 2 3 4 5
5	Apakah fungsi dan fitur yang ada pada sistem sudah memenuhi dengan yang diharapkan?	1 2 3 4 5
6	Apakah dengan adanya aplikasi LeaveRequest, proses transaksi permintaan dan perizinan cuti menjadi lebih mudah?	1 2 3 4 5



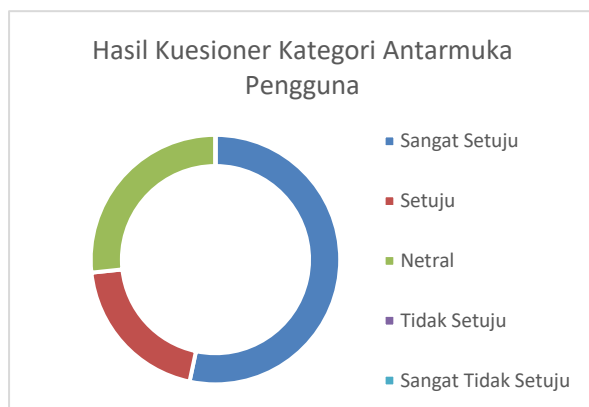
Tabel 3. Daftar Pertanyaan pada metode *Usability Testing* Kategori Antarmuka (*User Interface*)

No	Daftar Pertanyaan Skala (Kategori Kegunaan Sistem)	Penilaian
1	Apakah seluruh teks (warna, ukuran, dan jenis <i>font</i> ) pada aplikasi mudah dibaca dan dipahami?	1 2 3 4 5
2	Apakah seluruh simbol (warna, ukuran, penggambaran) mudah dilihat dan dimengerti?	1 2 3 4 5
3	Apakah letak posisi, pemilihan warna, dan ukuran pada tombol, tabel, panel, dan lain-lain sudah cocok?	1 2 3 4 5
4	Apakah menu dan fitur pada aplikasi mudah untuk diingat?	1 2 3 4 5
5	Apakah aplikasi mudah untuk dikenali, dioperasikan dan diingat?	1 2 3 4 5
6	Apakah tampilan pada aplikasi nyaman untuk dilihat dan tidak membosankan?	1 2 3 4 5

Kuesioner dibagikan kepada 30 responden yang terdiri dari karyawan dan mahasiswa peserta kerja praktek dari berbagai universitas. Berikut merupakan hasil dari rekapitulasi data yang digambarkan dalam bentuk diagram.



Grafik 1. Hasil Kuesioner Kategori Kegunaan Sistem (*System Usability*)



Grafik 2. Hasil Kuesioner Kategori Antarmuka Pengguna (*User Interface*)

Data yang didapat dari kuesioner kemudian diolah menggunakan skala Likert sehingga akan

menghasilkan *output* berupa sebuah informasi. Pengolahan data dengan skala Likert biasanya menggunakan lima skala pilihan dengan format sangat setuju, setuju, netral, tidak setuju dan sangat tidak setuju [12][13].

Tabel 4. Tabel Perhitungan Skala Likert Kategori Kegunaan Sistem (*System Usability*)

Jawaban	Jumlah	Nilai
Sangat Setuju (5)	16	80
Setuju (4)	8	32
Netral (3)	5	15
Tidak Setuju (2)	1	2
Sangat Tidak Setuju (1)	0	0
<b>Total Skor =</b>		<b>129</b>

Tabel 5. Tabel Perhitungan Skala Likert Kategori Antarmuka Pengguna (*User Interface*)

Jawaban	Jumlah	Nilai
Sangat Setuju (5)	16	80
Setuju (4)	6	24
Netral (3)	8	24
Tidak Setuju (2)	0	0
Sangat Tidak Setuju (1)	0	0
<b>Total Skor =</b>		<b>128</b>

Cara perhitungan untuk mendapatkan nilai dari kedua tabel tersebut adalah dengan cara mengalikan jumlah pemilih jawaban dengan bobot nilai dari masing-masing jawaban tersebut. Selanjutnya nilai tersebut dijumlahkan sehingga didapatkan total skor dari setiap tabel. Dapat dilihat dari tabel 4 dan 5 bahwa total skor yang didapatkan pada kategori Kegunaan Sistem (*System Usability*) adalah sebesar 129, dan pada kategori Antarmuka Pengguna (*User Interface*) sebesar 128.

Proses pengolahan data selanjutnya adalah mendapatkan skala persentase kepuasan pengguna terhadap aplikasi. Variabel yang perlu ditemukan adalah variabel *Y* yang merupakan jumlah total skor maksimum yang bisa didapatkan dengan rumus 1 [11].

$$Y = a x r \quad (1)$$

dengan *a* adalah bobot nilai tertinggi yaitu 5 poin dan *r* adalah jumlah responden yaitu sebanyak 30 orang. Setelah mendapatkan variabel *Y* maka bisa masuk ke rumus terakhir yaitu rumus menghitung persentase kepuasan pengguna. Rumusnya adalah sebagai berikut [11].

$$\% K = 100 x \frac{t}{Y} \quad (2)$$

dengan nilai *t* merupakan total skor yang didapat. Penggunaan skema perhitungan Likert tersebut bisa diimplementasikan sebagai berikut :

Variabel Y = 5 \* 30  
 Variabel Y = 150

% Kepuasan Kategori Kegunaan Sistem  
 = 100 \* (129 / 150)  
 = 100 \* 0,8600  
 = 86.00%

% Kepuasan Kategori Antarmuka Pengguna  
 = 100 \* (128/150)  
 = 100 \* 0.8533  
 = 85.33%

Supaya dapat mengetahui indeks hasil persentase yang didapatkan, maka berpacu pada tabel kelas interval yang ada pada Tabel 6. Pada tabel tersebut, dapat diketahui berapa level kepuasan pengguna dengan mencocokkan hasil persentase yang telah didapat dengan posisi indeks yang sudah tercantum[12].

Tabel 6. Tabel Indeks Kelas Interval Dengan Skala Likert

Kategori Kepuasan Pengguna	Interval
Sangat Tidak Setuju	0 % - 19.99 %
Tidak Setuju	20 % - 39.99 %
Netral	40 % - 59.99 %
Setuju	60 % - 79.99 %
Sangat Setuju	80 % - 100 %

Dilihat berdasarkan perhitungan menggunakan skema Likert tersebut, dapat disimpulkan bahwa: (1) persentase kepuasan pengguna pada kategori Kegunaan Sistem (*System Usability*) pada aplikasi LeaveRequest mencapai angka 86.00% sehingga dapat dikatakan bahwa rata-rata pengguna menyukai dan menyetujui penggunaan aplikasi LeaveRequest ini; (2) persentase kepuasan pengguna pada kategori Antarmuka Pengguna (*User Interface*) pada aplikasi LeaveRequest ini mencapai angka 85.33%, sehingga dapat dikatakan bahwa rata-rata pengguna menyukai dan menyetujui implementasi dari antarmuka pengguna yang ada pada aplikasi LeaveRequest ini.

#### 4. Kesimpulan

Aplikasi permintaan dan perizinan cuti LeaveRequest digunakan oleh pihak internal PT. XYZ guna menyesuaikan diri terhadap situasi pandemi Covid19 yang membatasi segala aktivitas secara tatap muka.

Dilihat berdasarkan tujuan penelitian ini yaitu membuat sebuah aplikasi berbasis *online* untuk menunjang proses permintaan cuti karyawan pada PT. XYZ, maka dapat disimpulkan bahwa aplikasi LeaveRequest ini telah berhasil mencapai tujuannya. Hal ini dibuktikan dengan

skor hasil kuesioner kategori kegunaan sistem yang mencapai 86.00% dan hasil kuesioner kategori antarmuka pengguna yang mencapai 85.33%.

Hasil dari adanya Aplikasi LeaveRequest ini, segala proses transaksi cuti menjadi lebih mudah, cepat, transparan, akurat dan aman. Aplikasi LeaveRequest membantu karyawan dalam proses membuat permintaan cuti karena aplikasi ini mudah diakses kapan dan dimana saja, sistem yang terstruktur membantu manajer dalam menyajikan data serta memproses perizinan cuti bawahannya. Lalu aplikasi ini juga memudahkan administrator dalam mencatat serta mengelola segala data seputar proses cuti di PT.XYZ.

#### Daftar Rujukan

- [1] D. Herdiana, "Implementasi Kebijakan Pembatasan Sosial Berskala Besar (Psbb) Sebagai Upaya Penanggulangan Corona Virus Disease 2019 (Covid-19)," *Jurnal Administrasi Publik*, vol. 2, no.2, pp. 2656-4939, 2020.
- [2] M. Hasrul, "Aspek Hukum Pemberlakuan Pembatasan Sosial Berskala Besar (Psbb) Dalam Rangka Penanganan Corona Virus Disease 2019 (Covid-19)," *legislatif*, vol. 3, no.2, p. 55, 2020.
- [3] INE, "Pengembangan Aplikasi Web Untuk Pengajuan Cuti Pegawai Secara Online," *Jurnal Algoritma*, vol. 12, no. 2, pp. 526-533, Dec. 2015.
- [4] A. Nurhadi and R. Sastra, "Pengembangan Aplikasi Web Pengajuan Cuti Pegawai Secara Online Studi Kasus : PT . Yapindo Transportama ( PCP Express )," *Bianglala Informatika*, vol. 5, no. 2, pp. 22–29, 2017.
- [5] M. Muhammad and V. Oktaviani, "Perancangan Sistem Informasi Cuti Karyawan Pada PT. Aneka Gas Industri Tbk Pekanbaru," *ITJRD*, vol. 3, no. 2, pp. 54 - 65, Feb. 2019.
- [6] A. Setiyanto, F. Samopa, and Alwi, "Pembuatan Sistem Informasi Cuti pada Kantor Pelayanan Perbendaharaan Negara dengan Menggunakan PHP dan MySQL," *Jurnal Teknik ITS*, vol. 2, no. 2, pp. 381–384, 2013.
- [7] R. Akbar and Indri, "Pembangunan Aplikasi Web Dengan Fitur Mobile Untuk Pada Kantor Wali Nagari Pagaruyung," *Prosiding Semnastek* vol. 4, no.6, pp. 1–2, 2017.
- [8] Yesputra, Rolly, Marpaung Nasrun, "Penerapan Arsitektur Model View Contoller (Mvc) Pada Sistem Informasi E-Skripsi Stmik Royal," *Jurnal Informatika Sains dan Teknologi*, vol. 3, no. 2, pp. 281–290, 2018.
- [9] P. Aliaga, "Implementasi Framework Spring Mvc Untuk Pembuatan Sistem Informasi Manajemen E Commerce," *Jurnal Universitas Sebelas Maret*, vol. 9, no. 1, pp. 76–99, 2010.
- [10] Rubiyanto, "Implementasi Role-Based Access Control (RBAC) Pada Pemanfaatan Data Kependudukan Ditingkat Kabupaten," *Prosiding Semnastek*, vol. 1, no. 1, pp. 1–10, 2017.
- [11] R. Somya, "Pengembangan Web Layanan Konsumen dan Penjaminan Mutu PT Pura Barutama Unit Boxindo dengan Framework Laravel dan Materialize" *AITI*, vol. 13, no. 1, pp. 15-30, September, 2017.
- [12] T. Nempung, T. Setyaningsih, and N. Syamsiah, "Otomatisasi Metode Penelitian Skala Likert Berbasis Web," *Prosiding Semnastek*, vol. 2, no. 4, pp. 1–8, 2015.
- [13] W. Budiaji, "Skala Pengukuran dan Jumlah Respon Skala Likert (The Measurement Scale and The Number of Responses in Likert Scale)," *Ilmu Pertan. dan Perikan.*, vol. 2, no. 2, pp. 127–133, 2013.