

BAB 2

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1. Tinjauan Pustaka

Penelitian ini menggunakan beberapa sumber pustaka sebagai referensi yang berhubungan dengan kasus atau metode yang akan diteliti, di antaranya yaitu:

Dalam penelitian oleh Bambang Kintoko (2019), yang berjudul “Aplikasi Pembuatan Surat Keluar Menggunakan Teknologi PHP dan *MySQL Improved* (Studi Kasus Fakultas Ilmu Agama Islam Universitas Islam Indonesia)”. Dalam penelitian tersebut meneliti tentang aplikasi yang digunakan untuk membuat permohonan surat dan pengarsipan secara digital untuk meningkatkan mutu pelayanan pada bidang layanan mahasiswa. Pengembangan perangkat lunak tersebut dilandasi akan banyaknya permintaan aplikasi yang berbasis *website* untuk menunjang kinerja pegawai dan layanan pada mahasiswa. Aplikasi ini dibangun dengan menggunakan teknologi web PHP dan *MySQLi* yang nantinya pengguna hanya perlu *link* dari *website* untuk *login* menggunakan *id* dan *password* yang sudah ditentukan pada sistem. Dengan adanya aplikasi pembuatan surat keluar pengarsipan dokumen menjadi lebih mudah dilakukan oleh petugas dan dapat melakukan penghematan kertas.

Emilia Alfina Raul (2021), melakukan penelitian dengan judul “Sistem Informasi Pengelolaan Surat Masuk dan Surat Keluar Berbasis Web di Kantor Dinas Sosial D.I. Yogyakarta”. Pada penelitian tersebut berfokus pada pengelolaan surat masuk dan surat keluar. Hal yang melandasi dalam penelitian tersebut karena

pada saat itu pengelolaan surat masih dalam bentuk fisik dan banyaknya penggunaan kertas serta pembuatan laporan harus dilakukan secara manual. Dalam sistem tersebut pembangunan aplikasi menggunakan bahasa pemrograman PHP dan menggunakan *data Base MySQL*. Dengan adanya aplikasi tersebut maka pengelolaan surat masuk dan keluar menjadi lebih mudah dilengkapi juga dengan adanya notifikasi *SMS Gateway*.

Sedangkan penelitian yang dilakukan oleh Andre Rizky Prasetya, Fadli Ranuharja, Ika Parma Dewi dan Syukhri (2022), tentang Perancangan Sistem Informasi Berbasis Web. Pada rancangan aplikasi tersebut difokuskan pada pelayanan masyarakat di kelurahan tanjung ayun sakti, di mana untuk masyarakat diharuskan melakukan *login* terlebih dahulu pada aplikasi. Dalam perancangan aplikasi tersebut menggunakan metode *waterfall*, sedangkan dalam pembangunan menggunakan bahasa pemrograman PHP dengan *framework CodeIgniter* dan XAMPP sebagai *data Base server*. Sistem aplikasi tersebut memiliki sepuluh aktor dengan masing-masing memiliki hak akses yang berbeda. Aplikasi tersebut dirancang untuk melakukan pelayanan kepada penduduk berupa pengajuan surat secara *online*.

Pada penelitian yang dilakukan oleh I Gusti Ngurah Swala Putra, I Putu Satwika, I Gede Juliana Eka Putra (2020), tentang Rancang Bangun Sistem Informasi Manajemen Administrasi Desa Berbasis *Web* Menggunakan *Framework* Laravel berfokus pada pengelolaan persuratan pada kantor Desa Sabda Blabatuh Gianyar. Hal yang menjadi landasan para peneliti tersebut karena selama ini pengelolaan persuratan masih dilakukan secara konvensional dan mengakibatkan

format surat menjadi berubah-ubah dan rawan terjadi kesalahan dalam pencatatan serta waktu menjadi tidak efisien. Pada pengembangan aplikasi tersebut menggunakan metode *Software Development Life Cycle* (SDLC) dengan model pengembangan *waterfall*. Sedangkan dalam pembangunan aplikasi menggunakan *framework* Laravel sebagai *backend* dan *framework* VueJs sebagai *frontend*. Hasil dari sistem yang dibangun dapat membantu dalam proses pembuatan surat menjadi lebih cepat serta format surat sudah menjadi ketetapan sehingga lebih akurat dan efisien.

Begitu pula dengan penelitian yang dilakukan oleh Masjudin, Cakra Adipura Wicaksana, Agusutrisno dan Rian Fahrizal (2022), tentang rancang bangun sistem informasi arsip dan persuratan pada program studi teknik elektro Untirta. Dalam penelitian tersebut difokuskan pada penyimpanan data dokumen kearsipan persuratan serta karya tulis ilmiah yang ada pada program studi teknik elektro Universitas Ageng Tirtayasa. Hal tersebut dilandasi akan budaya *smart and green* yang diterapkan pada Universitas Sultan Ageng Tirtayasa. Pada perancangan aplikasi menggunakan metode *waterfall* menggunakan bahasa pemrograman PHP dengan *framewrok* Laravel. Dengan adanya aplikasi tersebut pengarsipan dokumen menjadi lebih mudah dan juga menjadi lebih efisien dan efektif. Aplikasi tersebut juga mendukung penerapan budaya *paperless* yang sedang diterapkan di Universitas Sultan Ageng Tirtayasa.

Penelitian ini bertujuan untuk mengembangkan sistem informasi pengelolaan surat perijinan penelitian dan pengumpulan karya tulis ilmiah pada Politeknik Kesehatan Bhakti Setya Indonesia dalam upaya untuk memberikan solusi masalah

yang terjadi saat ini supaya dapat membantu dalam pelayanan kepada mahasiswa dalam pengelolaan surat perijinan studi pendahuluan maupun surat ijin penelitian serta dalam pengumpulan naskah atau dokumen karya tulis ilmiah dengan menggunakan sistem informasi yang disesuaikan dengan kebutuhan yang ada. Tabel perbandingan dengan penelitian sebelumnya akan ditunjukkan pada tabel 2.1.

Tabel 2.1. Perbandingan Tinjauan Pustaka

Penulis	Objek	Bahasa Pemrograman, Metode, Teknologi	Interface	Fitur
Kintoko (2019)	Fakultas Ilmu Agama Islam UII	PHP, <i>MySQL</i>	Web Browser	Permohonan surat. Pengarsipan.
Raul (2021)	Kantor Dinas Sosial DIY	PHP, <i>MySQL</i>	Web browser	Pengelolaan surat masuk dan keluar. Notifikasi <i>sms gateway</i>
Prasetya dkk., (2022)	Kelurahan Tanjung Ayun Sakti	PHP, <i>CodeIgniter, waterfall</i>	Web browser	Pengajuan surat. Sepuluh aktor dengan hak akses masing-masing
Putra dkk., (2020)	Kantor Desa Sabda Blabatuh Gianyar Bali	PHP, <i>Waterfall, Laravel, VueJs</i>	Web browser	Format surat baku. Pengelolaan persuratan.
Masjudi dkk., (2022)	Prodi Teknik Elektro Untirta	PHP, <i>waterfall, Laravel</i>	Web browser	Pengarsipan dokumen persuratan.
Dwi Suluh Pribadi	Politeknik Kesehatan Bhakti Setya Indonesia	PHP, <i>waterfall, Laravel</i>	Web browser	Notifikasi email mahasiswa. Cetak surat. Laporan dan pelacakan surat.

2.2. Dasar Teori

2.2.1. SDLC (*Software Development Life Cycle*)

SDLC pertama kali digunakan pada tahun 1960-an, untuk pengembangan sistem usaha berskala besar secara fungsional untuk membantu para konglomerat dimasa itu. SDLC atau *Software Development life Cycle* atau juga sering disebut *System Development Life Cycle* merupakan proses mengembangkan atau mengubah

suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya berdasarkan *best practice* atau cara-cara yang teruji dengan baik (Sukamto, 2022).

Dalam SDLC memiliki tahapan-tahapan secara global adalah sebagai berikut:

1. Inisiasi (*Initiation*)

Tahap ini ditandai dengan pembuatan proposal proyek perangkat lunak.

2. Pengembangan konsep sistem (*system concept development*)

Mendefinisikan lingkup konsep termasuk dokumen lingkup sistem, analisis, manfaat, biaya, manajemen rencana, dan pembelajaran kemudahan sistem.

3. Perencanaan (*planning*)

Mengembangkan rencana manajemen proyek dan dokumen perencanaan lainnya.

4. Analisis kebutuhan (*requirement analysis*)

Menganalisis kebutuhan dari pengguna sistem perangkat lunak (*user*) dan mengembangkan kebutuhan *user*. Membuat dokumen kebutuhan fungsional.

5. Desain (*design*)

Mentransformasikan kebutuhan detail menjadi kebutuhan yang lengkap, dokumen desain sistem berfokus pada bagaimana dapat memenuhi fungsi-fungsi yang dibutuhkan.

6. Pengembangan (*development*)

Melakukan konversi desain ke sistem informasi yang lengkap termasuk bagaimana memperoleh dan melakukan instalasi lingkungan sistem yang dibutuhkan; membuat basis data dan mempersiapkan prosedur kasus pengujian; mempersiapkan berkas pengujian, mengkodekan, pengkompilasian, memperbaiki serta membersihkan program; peninjauan pengujian.

7. Integrasi dan pengujian (*integration and test*)

Mendemonstrasikan sistem perangkat lunak bahwa telah memenuhi kebutuhan yang sesuai dengan spesifik pada dokumen kebutuhan fungsional.

8. Implementasi (*implementation*)

Implementasi perangkat lunak pada lingkungan produksi dan menjalankan resolusi dari permasalahan yang teridentifikasi dari fase integrasi dan pengujian.

9. Operasi dan pemeliharaan (*operation and maintenance*)

Mendeskripsikan pekerjaan untuk mengoperasikan dan memelihara pada lingkungan produksi.

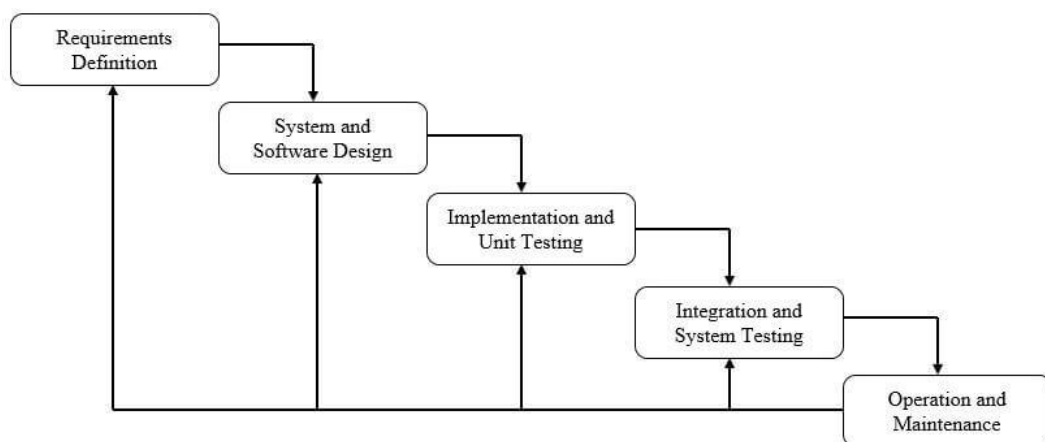
10. Disposisi (*disposition*)

Mendeskripsikan aktivitas akhir dari pengembangan sistem dan membangun data yang sebenarnya sesuai dengan aktivitas *user*.

SDLC mempunyai beberapa model, antara lain yaitu *waterfall*, *spiral*, *iterative*, *fountain*, *rapid prototyping*, *build and fix*, *synchronize and stabilize*, dan *extreme*.

2.2.2. Model *Waterfall*

Dalam pengembangan aplikasi *output* layanan administrasi persuratan ini menggunakan SDLC model *waterfall*. Model *waterfall* sering juga disebut dengan model klasik karena bersifat sistematis, berurutan dalam pembangunan *software*. Model ini termasuk ke dalam model generik pada rekayasa perangkat lunak dan pertama kali dikenalkan oleh Winston Royce pada sekitar tahun 1970, sehingga sering kali dianggap kuno, tetapi merupakan model yang paling banyak digunakan dalam *Software Engineering* (SE). Model *waterfall* menggunakan pendekatan sekuntial atau disebut berurutan dimulai dari analisis, desain, mengkodekan, pengujian, dan tahap pendukung (Sukamto, 2022). Sedangkan menurut Pressman (2015), disebut sebagai model *waterfall* karena tahap demi tahap yang dilalui harus berurutan atau tahap yang dilalui harus menunggu tahap sebelumnya selesai. Jad menurut penulis pengembangan model *waterfall* merupakan pengembangan di mana setiap proses harus dilakukan terlebih dahulu sampai dengan selesai baru bisa melakukan ke tahap selanjutnya atau dengan kata lain secara prosedural. Perancangan model *waterfall* dapat ditunjukkan pada gambar 2.1.



Gambar 2.1. Model *Waterfall*

Dalam pengembangan model *Waterfall* memiliki tahapan sebagai berikut:

1. Analisis Kebutuhan

Proses menganalisis dan pengumpulan kebutuhan sistem secara intensif untuk menyespesifikasikan kebutuhan perangkat lunak serta kebutuhan dari *user*.

2. Desain Sistem dan *Software*

Proses desain yang fokus pada desain pembuatan program termasuk struktur data, arsitektur perangkat lunak, representasi antar muka dan prosedur mengkodekan.

3. Mengkodekan

Menerjemahkan desain ke dalam bahasa yang dimengerti oleh komputer sesuai dengan desain yang telah dibuat pada tahapan desain.

4. Pengujian

Pengujian perangkat lunak dari segi logika dan fungsional serta memastikan bahwa semua bagian sudah diuji.

5. Operasi dan Pemeliharaan

Pemeliharaan sistem sebagai perbaikan ketika mengalami perubahan saat sudah digunakan oleh *user* atau setelah dioperasikan.

2.2.3. HTML 5

Menurut Powell (2010), *Hypertext Markup Language* atau HTML adalah bahasa *markup*. Bahasa *markup* adalah bahasa yang digunakan untuk menerjemahkan informasi dari web browser tentang struktur halaman, konten dan sebagainya. Jadi, HTML adalah sebuah dokumen teks yang berisi informasi dari

web browser. Perintah-perintah yang ada di dokumen HTML diawali dengan tag pembuka `<html>` dan diakhiri dengan tag penutup `</html>`. Bahasa pemrograman web terus mengalami perkembangan. Versi yang terbaru adalah HTML5. Kessin (2012) menyatakan bahwa HTML5 membuat kelas-kelas di dalam web sehingga aplikasi yang dibuat terlihat lebih nyata. HTML5 didukung dengan Javascript, API browser, AJAX sehingga lebih mudah dalam pembuatan aplikasi. HTML5 membuat aplikasi web lebih interaktif pada browser dari sisi pengguna dengan adanya fitur-fitur multimedia dan grafis.

2.2.4. CSS 3

Menurut Mansfield (2005), Cascading Style Sheet atau CSS adalah Bahasa pemrograman yang ditanam pada halaman HTML dan berjalan pada browser. CSS memungkinkan desain halaman web terlihat lebih baik dan efektif. CSS memungkinkan pemisahan desain dengan konten ketika aplikasi web dibangun karena keterbatasan HTML mengelola tampilan. CSS mengelola desain web menjadi halaman visual yang menarik. Dokumen CSS dapat terpisah dari halaman HTML dan memiliki ekstensi `*.css`. McFarland (2013) menyatakan bahwa CSS menawarkan beberapa keuntungan, yaitu menampilkan tampilan yang lebih baik dari HTML, menghemat tag yang digunakan di dalam dokumen HTML untuk pengaturan tampilan, mempermudah proses memperbaiki tampilan dengan dokumen CSS yang terpisah. Menurut Powers (2012), CSS3 adalah versi terbaru dari CSS. CSS3 terdiri dari beberapa modul yang menawarkan fitur baru, yaitu CSS3 selektor, warna, nilai dan unit, media, *background*, teks, huruf, gradien, multikolom, transformasi, transisi, animasi dan *flexibel* Box.

2.2.5. PHP

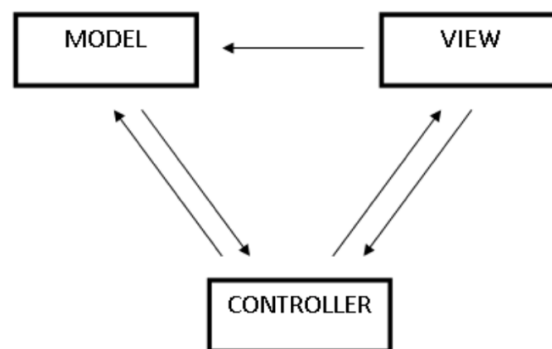
Menurut Nixon (2018), *Hypertext Preprocessor* (PHP) adalah bahasa pemrograman yang digunakan untuk membuat server menghasilkan keluaran yang dinamis dan ditanam pada bahasa HTML. Dalam penggunaannya, halaman web dihasilkan dengan kombinasi dari bahasa PHP, HTML, Javascript dan beberapa perintah *MySQL*. PHP menggunakan beberapa jenis tag, tag pembuka dan penutup yang standar adalah `<?php ?>`. Setiap perintah pada dokumen PHP harus diakhiri dengan tanda titik koma (.). Dokumen PHP disimpan dengan ekstensi *.php. Dokumen PHP dapat mengambil dokumen HTML seperti index.html disimpan dengan index.php dan menghasilkan halaman web yang sama persis dengan aslinya.

2.2.6. MySQL

MySQL merupakan sebuah sistem manajemen data Base yang bersifat open *source*. Menurut Robin Nixon (2018), *MySQL* merupakan basis data yang paling banyak digunakan dalam pemrograman web dengan bahasa pemrograman PHP. SQL pada *MySQL* sendiri adalah *Structured Query Language*, yaitu bahasa terstruktur yang menjelaskan permintaan untuk mengakses dan mengelola basis data. *MySQL* sangat *scalable*, yang berarti dapat berkembang sesuai aplikasi web yang dikembangkan atau dibuat. Tujuan dari dibuatnya *MySQL* adalah untuk memungkinkan permintaan sederhana dari basis data melalui perintah DDL maupun DML.

2.2.7. Konsep MVC

Model-View-Controller (MVC) adalah model pembuatan program yang menerapkan arsitektur aplikasi menjadi tiga bagian, yaitu: memisahkan antara proses, tampilan dan bagian yang menghubungkan dengan basis data. MVC mempunyai tujuan untuk memisahkan proses bisnis dari pertimbangan antarmuka pengguna agar para pengembang aplikasi bisa lebih mudah mengembangkan salah satu bagian sehingga tidak memberikan pengaruh pada bagian yang lain (Badiyanto, 2013). Jika digambarkan maka alur kerja MVC akan tampak seperti pada gambar 2.2:



Gambar 2.2. Konsep MVC

Badiyanto (2013) menjelaskan tentang *Model-View-Controller* sebagai berikut:

1. Model

Model merupakan kelas yang mendasari dari logika proses dalam aplikasi perangkat lunak dan kelas yang terkait dengannya. Model adalah suatu objek yang tidak mengandung informasi tentang *user interface*. Model juga merupakan suatu kelas yang berisi metode atau fungsi yang digunakan untuk menyimpan data dan aturan bisnis yang relevan.

2. *View*

View merupakan kumpulan kelas dari kelas yang mewakili unsur-unsur dalam antarmuka, dalam *view* terdapat nama yang dipakai untuk mengidentifikasi *file script* tampilan saat dipanggil melalui fungsi *render*. Nama *view* sama seperti nama *file script view-nya*.

3. *Controller*

Controller merupakan kelas yang menghubungkan model dan *view*, digunakan untuk berkomunikasi antara kelas dalam model dan *view*. *Controller* mempunyai *action* standar, ketika permintaan pengguna tidak menetapkan *action* mana yang dijalankan, maka program akan menjalankan *action* standar.

2.2.8. **Laravel**

Laravel merupakan sebuah *framework* PHP, *framework* sendiri merupakan sebuah kumpulan kode program siap pakai dengan aturan penulisan tertentu yang bertujuan untuk memudahkan serta mempercepat pembuatan sebuah aplikasi. Awal dari ide pembuatan *framework* Laravel berasal dari masalah yang dihadapi oleh Taylor Otwell pada tahun 2011. Secara resmi Laravel versi beta dirilis ke publik pertama kali pada 9 Juni 2011, yang diikuti dengan Laravel 1.0. sebulan sesudahnya. Sejak dirilis, kalangan *open source* tertarik untuk melakukan pengembangan Laravel agar lebih lengkap. Dalam pengembangan Laravel, tim pengembang memutuskan untuk membuat *update* terjadwal setiap 6 bulan sekali, yang kemudian menjadi 1 tahun sekali sejak diluncurkannya Laravel 8 (Pratama, 2022).

Sedangkan menurut Aminudin (2015), “Laravel adalah sebuah *Framework* PHP yang dirilis di bawah lisensi MIT dengan kode sumber yang sudah disediakan oleh *Github*, sama seperti *framework* yang lain, Laravel dibangun dengan konsep MVC (*Model-View-Controller*), kemudian Laravel dilengkapi juga *command line tool* yang bernama “*Artisan*” yang bisa digunakan untuk *packaging*, *bundle* dan instalasi *bundle* melalui *command prompt*”.

Laravel memiliki ukuran *file* yang sangat besar dan itu menjadi salah satu kekurangan. Dalam Laravel terdapat *file* yang bersifat *default* yaitu *vendor*, *direktori* tersebut tidak bisa sembarangan dihapus karena *file* dalam *direktori* tersebut digunakan untuk menyimpan seluruh *dependency manager* yang digunakan (Yudhanto & Adi Prasetyo, 2019). Berikut adalah dasar-dasar *Laravel*:

1. Artisan

Artisan adalah *command line* atau perintah yang dijalankan melalui terminal dan disediakan beberapa perintah yang dapat digunakan selama melakukan pengembangan dan pembangunan sebuah aplikasi. Salah satu fungsi dari *php artisan* ialah “*php artisan serve*” di mana fungsi tersebut dapat digunakan untuk menjalankan *website* yang telah dibuat tanpa menjalankan *web server* lokal.

2. Migration

Migration merupakan sebuah fitur yang menyediakan cara baru dalam membuat *database*. Dengan menggunakan *migration* cara membuat *database* melalui *Command Line Interface (CLI) database* atau dengan menggunakan aplikasi *database manager* digantikan dengan menggunakan *class*. Tahapan

dalam menggunakan *migration* adalah membuat *class* kemudian melakukan perintah migrate melalui *command line interface* (CLI) *artisan*.

3. Routing

Routing merupakan suatu proses yang memiliki tujuan agar suatu item yang diinginkan dapat sampai ke tujuan. Dengan menggunakan routing maka dapat ditentukan halaman yang akan muncul ketika diakses oleh *user*.

4. Controller

Controller adalah suatu proses yang bertujuan untuk mengambil permintaan, menginisialisasi, memanggil *model* untuk diteruskan ke *view*.

5. View (blade templating)

Blade merupakan *templating engine* bawaan dari laravel. *Blade* memiliki kode-kode yang lebih mudah untuk menghasilkan sebuah tampilan aplikasi. Pada dasarnya *blade* merupakan *view* tetapi dengan menggunakan blade maka akan mempermudah untuk mengatur tampilan website dengan menampilkan data. Cara untuk membuat *file view* menjadi *file blade* dengan menambahkan ekstensi *.blade.php* pada file view. Untuk memanggil file blade sama dengan cara memanggil file view biasa.

6. Middleware

Middleware merupakan penengah antara *request* yang masuk dengan *controller* yang akan dituju atau dengan kata lain merupakan sebuah fitur yang menyediakan mekanisme untuk melakukan filter HTTP *request* yang akan masuk ke aplikasi.

7. Session

Session adalah sebuah cara yang digunakan untuk menyimpan pada *server* dan penyimpanan tersebut akan digunakan pada beberapa halaman termasuk dalam halaman itu sendiri. Dalam penggunaannya *session* dapat digunakan dalam dua cara. Cara pertama *session* dibuat menggunakan *Request* dan cara kedua digunakan sebagai fungsi *global helper session*.

2.2.9. Notifikasi dan Email

Menurut KBBI (Kamus Besar Bahasa Indonesia), notifikasi adalah pemberitahuan atau kabar tentang penawaran barang dan sebagainya. Notifikasi yang berkaitan dengan sebuah sistem dapat diartikan sebuah pemberitahuan yang diberikan oleh sistem kepada pengguna baik melalui email, ponsel, maupun internet. Notifikasi dapat berupa pemberitahuan yang berisi teks kata, gambar, video, maupun suara. Apabila pengguna mengatur sebuah sistem atau aplikasi untuk mengeluarkan sebuah notifikasi, maka notifikasi tersebut akan masuk terlebih dahulu sebagai ikon dalam area notifikasi. Untuk melihat detail pemberitahuan, pengguna membuka laci pemberitahuan. Baik area pemberitahuan maupun laci pemberitahuan adalah area-area yang dikontrol sistem yang bisa dilihat pengguna kapan saja.

Email atau surat elektronik merupakan suatu sarana untuk mengirimkan dan menerima surat atau pesan dengan format digital melalui jalur jaringan komputer dan internet. Email dapat dibuat dan dikirimkan ke alamat email lain dengan menggunakan komputer dan gadget seperti *smartphone* yang terhubung dengan internet. Email juga sebagai identitas diri di internet di mana orang lain dapat menghubungi pengguna email melalui alamat email yang dimiliki