

**MEMBANGUN APLIKASI *TEMPLATE* PERSURATAN  
MENGUNAKAN *RESTFUL* API MANAJEMEN  
PERSURATAN UNIVERSITAS HASANUDDIN**

**SKRIPSI**



**ANDI FIQRIAH AMALIAH**

**H131 15 512**

**PROGRAM STUDI ILMU KOMPUTER DEPARTEMEN MATEMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS HASANUDDIN**

**MAKASSAR**

**NOVEMBER 2019**



**MEMBANGUN APLIKASI *TEMPLATE* PERSURATAN  
MENGUNAKAN *RESTFUL* API MANAJEMEN  
PERSURATAN UNIVERSITAS HASANUDDIN**



**SKRIPSI**

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Ilmu Komputer pada Program Studi Ilmu Komputer Departemen Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin Makassar

**ANDI FIQRIAH AMALIAH**

**H13 115 512**

**PROGRAM STUDI ILMU KOMPUTER DEPARTEMEN MATEMATIKA**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

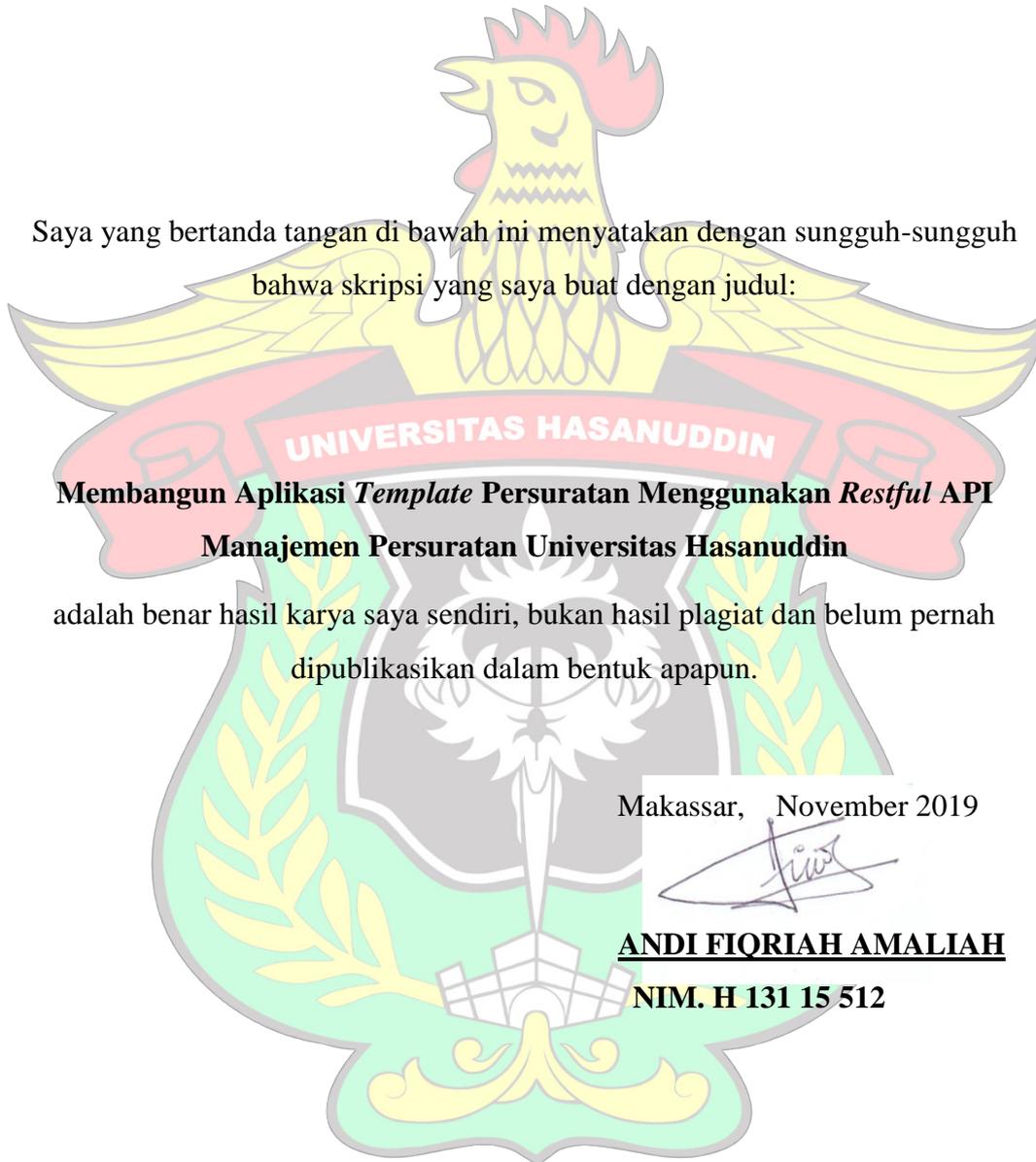
**UNIVERSITAS HASANUDDIN**

**MAKASSAR**

**NOVEMBER 2019**



## LEMBAR PERNYATAAN KEOTENTIKAN



Saya yang bertanda tangan di bawah ini menyatakan dengan sungguh-sungguh bahwa skripsi yang saya buat dengan judul:

**Membangun Aplikasi *Template* Persuratan Menggunakan *Restful* API  
Manajemen Persuratan Universitas Hasanuddin**

adalah benar hasil karya saya sendiri, bukan hasil plagiat dan belum pernah dipublikasikan dalam bentuk apapun.

Makassar, November 2019



**ANDI FIORIAH AMALIAH**

**NIM. H 131 15 512**



**MEMBANGUN APLIKASI *TEMPLATE* PERSURATAN  
MENGUNAKAN *RESTFUL* API MANAJEMEN  
PERSURATAN UNIVERSITAS HASANUDDIN**

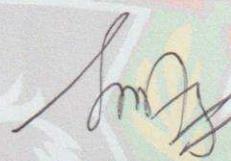
Disetujui oleh :

Pembimbing Utama

Pembimbing Pertama



**Dr. Hendra, S.Si., Kom.**  
NIP. 197601022002121001



**Supri Bin Hj Amir, S.Si, M.Eng**  
NIP. 19880504 201903 1 012

Pada tanggal, November 2019



## HALAMAN PENGESAHAN

Skripsi ini diajikan oleh:

Nama : Andi Fiqriah Amaliah  
NIM : H131 15 512  
Progam Studi : Ilmu Komputer  
Judul Skripsi : Membangun Aplikasi *Template* Persuratan  
Menggunakan *Restful* API Manajemen  
Persuratan Universitas Hasanuddin

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Sains Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin.

### DEWAN PENGUJI

#### Tanda Tangan

1. Ketua : Dr. Hendra, S.Si., M.Kom. (.....)
2. Sekretaris : Supri Bin Hj Amir, S.Si, M.Eng (.....)
3. Anggota : Dr. Diaraya, M.Ak (.....)
4. Anggota : Dr. Muhammad Hasbi, M.Sc (.....)

Ditetapkan di : Makassar

Tanggal :



## KATA PENGANTAR

*Assalamu'alaikum Warohmatullahi Wabarokatuh.*

AlhamdulillahRobbila'lamin, segala puji senantiasa dipanjatkan untuk menyampaikan rasa syukur kepada **Allah Subhanahu Wata'ala** atas segala limpahan rahmat dan inayah yang diberikan kepada penulis sehingga dapat menyelesaikan penulisan skripsi dengan judul **“Membangun Aplikasi Template Persuratan Menggunakan Restful API Manajemen Persuratan Universitas Hasanuddin”** sebagai salah satu syarat untuk memperoleh gelar Sarjana Sains pada Program Studi Ilmu Komputer Departemen Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin Makassar. Salam dan sholawat InsyaAllah senantiasa tercurah kepada **Nabi Muhammad Shallallahu'alaihi Wasallam** yang telah membawa umat Islam dari alam kegelapan ke alam yang terang – menderang.

Penulis menyadari bahwa penyelesaian tugas akhir ini tidak luput dari bantuan dan dukungan dari berbagai pihak. Oleh karena itu, penulis ucapkan terima kasih yang sebesar-besarnya kepada kedua orang tua penulis, Ayahanda **Andi Muntahar** dan Ibunda **Siti Suriani** atas didikan dan cinta kasih serta doa dan nasehat yang selalu setia diberikan kepada penulis. Penulis juga mengucapkan terima kasih yang sedalam-dalamnya kepada seluruh keluarga besar penulis atas doa dan dukungan yang selalu diberikan selama ini.

Penghargaan yang tulus dan ucapan terima kasih dengan penuh keikhlasan juga penulis ucapkan kepada:

1. **Rektor Universitas Hasanuddin, Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam, Bapak Prof. Dr. Amir Kamal Amir, M.Sc** selaku Ketua Departemen Matematika, **segenap dosen pengajar, staf departemen Matematika, dan staf Fakultas MIPA**, yang telah membekali ilmu dan kemudahan-kemudahan kepada penulis dalam berbagai hal selama menjadi mahasiswa di jurusan Matematika.

**Hendra, S.Si., M.Kom** selaku dosen pembimbing utama atas nasehat, bimbingan, do'a, dan kesabaran dengan setulus hati telah meluangkan waktunya membimbing penulis hingga akhir.



3. **Supri Bin Hj Amir, S.Si, M.Eng,** selaku dosen pembimbing pertama yang dengan penuh kesabaran senantiasa memberikan saran dan kritikan dalam penulisan tugas akhir ini.
4. **Dr. Diaraya, M.Ak.** selaku ketua penguji yang penuh perhatian telah memberikan kritik dan saran yang membangun untuk penulisan tugas akhir ini.
5. **Dr. Muhammad Hasbi, M.Sc.** selaku sekretaris penguji yang selama *seminar* telah banyak memberikan kritik dan saran yang sangat berharga dalam memperbaiki skripsi ke arah yang lebih baik.
6. Terimakasih untuk yang tersayang sahabat ilmu komputer terkhusus angkatan 2015 yang telah menjadi sahabat terbaik selama menyandang status mahasiswa dan ikut turut meramaikan dan membantu kegiatan-kegiatan ilmu komputer, serta senantiasa memberikan dukungan dalam menyelesaikan tugas akhir dan yang tercinta junior manis dan tampan yang membuat kuliah menjadi lebih berwarna.
7. **Bima Satria Yudha Mohammad, S.Si , Adrianto Saputra Mansur, S.Si , Muh. Emir Toripuji, Andi Nirwana, Irma, Ummaerah S, Agreani Mangera T, Rahmawati, Wahyu Saputra B., Andi Muhammad Asdar, Adrian Saputra, Hedrianto, Baish Fajar Luthfi, Ahmad Fikri Aslam Suharto, Muh. Fachrul Riza, Thomy Pongsongga, Arjudi Antonius, Achmad Muchtadin, Khusnul Hatima Suryani, Muhammad Anwar Sadad, Windi Reza Pratiwi, Faathir Anshary, Asmila, Gidion P, Muh. Hasbi Hasnan Habib, Ilham, Suhastina, Putry Qarynah, Sinar Nur Insani, Ayu Lestari, Amaliyah Afifah, Ikhsan, Jems, Reskyawan**  
Terimakasih untuk sahabat tercinta yang setia mendampingi dari awal perkuliahan sampai sekarang,terimakasih untuk semua kegiatan yang telah kita lalui bersama, terimakasih untuk semua bantuannya, terimakasih untuk canda tawa dan terimakasih untuk kekompakannya.
8. **Andi Rahmat Rilangi** selaku adik kandung yang menjadi alasan untuk segera menyelesaikan perkuliahan lebih cepat .
9. **Atharazzka** selaku anak dari **Danti Hasan** yang menjadi motivasi segera menyelesaikan skripsi agar bisa bertemu.



10. **Andi Nurul Ilmi, Nur fatimah Yuliani, Rany Nursyafitri, Yanti Boli, Wardah Nurdin, Ade Irma Safitri, Alfiandhani Suci M, Nur Asryany Pratama, S.KM , Zulkifli Muhammad, Ilham Taha, Andi Baso T.A, Mega Hasnawi** terimakasih untuk yang tercinta atas semua dukungan dan doa nya.
11. Kepada semua pihak yang tidak dapat penulis sebutkan satu-persatu, semoga segala dukungan dan partisipasi yang diberikan kepada penulis bernilai ibadah di sisi **Allah SWT**.

Akhir kata semoga tulisan ini dapat memberikan manfaat bagi para mahasiswa, khususnya bagi Mahasiswa Fakultas Matematika dan Ilmu Pengetahuan Alam jurusan Matematika dan bagi Perguruan Tinggi.

*Wassalamu'alaikum Warohmatullahi Wabarokatuh.*

Makassar, November 2019



Andi Fiqriah Amaliah



**PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK  
KEPENTINGAN AKADEMIS**

---

---

Sebagai sivitas akademik Universitas Hasanuddin, saya yang bertanda tangan di bawah ini:

Nama : Andi Fiqriah Amaliah  
NIM : H131 15 512  
Program Studi : Ilmu Komputer  
Departemen : Matematika  
Fakultas : Matematika dan Ilmu Pengetahuan Alam  
Jenis karya : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Hasanuddin **Hak Prediktor Royalti Noneksklusif (*Non-exclusive Royalty- Free Right*)** atas tugas akhir saya yang berjudul:

**“Membangun Aplikasi *Template* Persuratan Menggunakan *Restful API*  
Manajemen Persuratan Universitas Hasanuddin”**

Beserta perangkat yang ada (jika diperlukan). Terkait dengan hal di atas, maka pihak universitas berhak menyimpan, mengalih-media/format-kan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di Makassar pada tanggal, November 2019

Yang menyatakan



(Fiqriah Amaliah)

## ABSTRAK

Manajemen persuratan Universitas Hasanuddin hanya dipakai untuk mengambil nomor surat dan belum ada modul yang disediakan untuk mencetak surat secara otomatis. Dari permasalahan tersebut memunculkan gagasan untuk membangun suatu aplikasi template surat otomatis berbasis *web*, aplikasi ini menggunakan *Codeigniter* untuk membangun aplikasi, HTML dan CSS sebagai bahasa markup yang digunakan untuk membuat halaman web dari aplikasi, PHP digunakan sebagai bahasa pemrograman dari aplikasi ini, dan *Restful API* sebagai penghubung antara aplikasi template persuratan dengan *database* kampus untuk mengambil nomor surat secara otomatis, sehingga *admin* hanya mengisi *form* yang diperlukan saat pembuatan surat di aplikasi *template* persuratan dan menghasilkan cetakan berbentuk pdf yang dapat di *download*.

Kata Kunci : Aplikasi Persuratan, *restful API*, *codeigniter*, HTML, CSS dan PHP



**ABSTRACT**

Hasanuddin University correspondence management is only used to retrieve letter numbers and there is no module provided for printing letters automatically. From these problems led to the idea of building a web-based automated letter template application, this application uses Codeigniter to build applications, HTML and CSS as markup languages used to create web pages from applications, PHP is used as the programming language of this application, and the Restful API as a liaison between the correspondence template application with the campus database to retrieve letter numbers automatically, so the admin only fills in the forms needed when making letters in the correspondence template application and produces print-out pdfs that can be downloaded.

Keywords: Correspondence application, restful API , codeigniter , HTML , CSS and PHP

Judul : Membangun Aplikasi *Template* Persuratan Menggunakan *Restful* API Manajemen Persuratan Universitas Hasanuddin

Nama : Andi Fiqriah Amaliah

: H131 15 512

Studi : Ilmu Komputer



DAFTAR ISI

**HALAMAN JUDUL** ..... i

**LEMBAR PERNYATAAN KEOTENTIKAN** ..... ii

**HALAMAN PERSETUJUAN PEMBIMBING** ..... iii

**HALAMAN PENGESAHAN** ..... iv

**KATA PENGANTAR** ..... v

**PUBLIKASI KARYA ILMIAH** ..... viii

**ABSTRAK** ..... ix

    ABSTRACT ..... x

**DAFTAR ISI** ..... xi

**DAFTAR GAMBAR** ..... xiii

**BAB I PENDAHULUAN** ..... 1

    1.1 Latar Belakang ..... 1

    1.2 Rumusan Masalah ..... 2

    1.3 Tujuan ..... 2

    1.4 Manfaat ..... 2

    1.5 Batasan Masalah ..... 2

**BAB II TINJAUAN PUSTAKA** ..... 3

    2.1 Landasan Teori ..... 3

        2.1.1 Surat Resmi ..... 3

        2.1.2 Desain Template ..... 7

        2.1.3 Web Aplikasi ..... 7

        2.1.4 Restful API ..... 9

        2.1.5 PHP ..... 11

        2.1.6 *CodeIgniter* ..... 12

        2.1.7 *My SQL* ..... 13

    2.2 Kerangka Pikir ..... 16

**BAB III METODE PENELITIAN** ..... 17

    3.1 Tahapan Penelitian ..... 17

        Rancangan Sistem ..... 18

        Use Case Diagram ..... 18

        Waktu dan Lokasi Penelitian ..... 18



3.4 Sumber Data .....18

3.5 Instrumen Penelitian .....19

**BAB IV HASIL DAN PEMBAHASAN ..... 20**

4.1 Tahapan Penelitian .....20

4.1.1 *Class Diagram*.....20

4.1.2 ERD .....20

4.1.3 *Acitivity Diagram* .....21

4.1.4 *Sequence Diagram*.....22

4.1.5 *Deployment Diagram* .....24

4.1.6 Implementasi MVC (*Model,View,Controller*).....24

4.1.7 *Restful API*.....28

**BAB V KESIMPULAN DAN SARAN ..... 31**

5.1 Kesimpulan.....31

5.2 Saran .....31

**DAFTAR PUSTAKA ..... 32**



DAFTAR GAMBAR

Gambar 1. Cara kerja Restful API ..... 9

Gambar 2. Alur Kerja Framework CodeIgniter ..... 13

Gambar 3. Kerangka pikir..... 16

Gambar 4. Flowchart..... 17

Gambar 5. Use Case Diagram ..... 18

Gambar 6. Class diagram ..... 20

Gambar 7. Entity Relationship Diagram ..... 21

Gambar 8. Activity diagram tampilkan buat surat ..... 21

Gambar 9. Activity diagram simpan data surat..... 22

Gambar 10. Sequence diagram halaman login..... 22

Gambar 11. Sequence diagram tampilkan halaman buat surat (form inputan).... 23

Gambar 12. Sequence Diagram simpan data surat ..... 23

Gambar 13. Deployment diagram aplikasi template persuratan ..... 24

Gambar 14. Model view controller ..... 24

Gambar 15.Struktur folder model aplikasi template persuratan ..... 25

Gambar 16. Contoh sintaks model ..... 25

Gambar 17. Struktur folder view ..... 26

Gambar 18. Contoh sintaks view ..... 26

Gambar 19. Struktur folder controller..... 27

Gambar 20. Contoh sintaks controller ..... 27

Gambar 21. Sintaks login..... 28

Gambar 22. Sintaks buat method post ..... 28

Gambar 23. Sintaks query ..... 28

Gambar 24. Sintaks HTTP request ..... 28

Gambar 25. Sintaks mengambil variabel ..... 28

Gambar 26. Sintaks mengkonversi variabel..... 29

Gambar 27. Sintaks memasukkan token ..... 29

Gambar 28. Sintaks menginisiasi url ..... 29

Gambar 29. Sintaks request..... 29

Gambar 30. Sintaks eksekusi url..... 29

Gambar 31. Sintaks mengirim request ..... 29

Gambar 32. Sintaks mengambil variabel dari HTTP ..... 29



## BAB I PENDAHULUAN

### 1.1 Latar Belakang

Persuratan menjadi hal yang tak terlepas dari aktivitas keseharian instansi dan organisasi surat-menyurat menjadi media komunikasi tertulis antar instansi ataupun masyarakat, sehingga surat masih menjadi rujukan utama pada setiap aktivitas di instansi. seiring dengan perkembangan zaman saat ini persuratan telah berkembang dengan berbagai macam kecanggihan dan kemudahannya.

Meskipun saat ini telah ada fasilitas teknologi seperti email (surat elektronik) dan lainnya, peran surat resmi dalam bentuk *print out* masih tetap diperlukan untuk keperluan tertentu, seperti keperluan dalam manajemen persuratan kampus.

Masalah yang sering terjadi saat ini yaitu adalah manajemen persuratan kampus secara umum masih berjalan secara manual, termasuk dengan manajemen persuratan Universitas Hasanuddin, lebih tepatnya di Fakultas Matematika dan Ilmu Pengetahuan Alam Departemen Matematika, dalam manajemen persuratan Departemen Matematika masih berjalan secara manual, karena harus mengetik dan mengecek satu per satu surat yang akan dibuat. Sementara hal tersebut tidak seimbang dengan intensitas persuratan yang semakin tinggi dan menuntut kecepatan waktu, kecepatan menjadi hal sensitif dalam kinerja instansi dan manajemen kampus. Mahasiswa telah merubah pola pandangnya, semakin cepat pengurusan persuratan dalam kampus maka kinerja instansi tersebut semakin baik. Hal ini tentu menjadi sebuah sugesti bagi pegawai manajemen persuratan kampus untuk meningkatkan efisiensi kinerjanya.

Selaras dengan masalah di atas dapat disimpulkan bahwa solusi dari permasalahan ini adalah dengan membangun sebuah aplikasi template surat otomatis yang macam-macam suratnya telah tersimpan dan masuk dalam aplikasi template surat tersebut menggunakan *Restful API* yang menjadi penghubung aplikasi template dan *database* web manajemen kampus, karena manajemen persuratan Universitas Hasanuddin hanya dipakai untuk mengambil



nomor surat dan belum ada modul yang disediakan untuk mencetak surat secara otomatis.

## 1.2 Rumusan Masalah

Berdasarkan uraian di atas maka yang menjadi rumusan masalah dalam penelitian ini yaitu Mengembangkan Aplikasi Persuratan menggunakan *Restful API* pada Manajemen Persuratan Universitas Hasanuddin

## 1.3 Tujuan

Tujuan dari penelitian ini adalah sebagai berikut:

1. Menghasilkan dokumentasi surat yang terkoneksi dengan Sistem Informasi persuratan Universitas Hasanuddin
2. Memudahkan pembuatan surat Departemen Matematika

## 1.4 Manfaat

Hasil dari penelitian ini diharapkan dapat memberikan manfaat dengan adanya fitur tambahan dalam *website* manajemen persuratan Universitas Hasanuddin ini pegawai jurusan tidak lagi membuat surat secara manual dan dapat mempercepat pekerjaan dalam pengurusan surat menyurat dalam kampus.

## 1.5 Batasan Masalah

Adapun batasan masalah yang dilakukan agar pengerjaan tugas akhir ini tidak terlalu luas. Maka masalah yang akan dibahas dibatasi pada:

1. Hanya dilakukan pada manajemen persuratan Departemen Matematika Universitas Hasanuddin.
2. Aplikasi dibangun menggunakan *Framework Codeigniter*
3. *Database* yang digunakan adalah *MySQL*
4. Surat yang dibuat adalah Surat Ralat Judul Skripsi dan Surat Usulan Tim Penguji



## BAB II TINJAUAN PUSTAKA

### 2.1 Landasan Teori

#### 2.1.1 Surat Resmi

Pengertian Surat Resmi adalah surat yang digunakan untuk keperluan formal/ resmi oleh pihak-pihak tertentu, baik itu perorangan, lembaga, organisasi, atau instansi tertentu untuk saling berkomunikasi satu sama lain secara formal.

Surat resmi dibuat dengan kaidah dan aturan yang telah ditentukan, misalnya penggunaan bahasa baku, isi surat harus efektif dan jelas, dan dibuat dengan cermat sesuai keperluannya.

##### 1. Fungsi surat resmi

Surat resmi memiliki fungsi khusus yang tidak dapat ditemukan pada jenis surat lainnya. Mengacu pada pengertian surat resmi di atas, berikut ini adalah beberapa fungsi surat resmi tersebut:

- Sebagai sarana informasi atau pemberitahuan mengenai hal-hal tertentu yang disampaikan oleh satu pihak kepada pihak lainnya, misalnya penyampaian gagasan atau pemikiran.
- Surat resmi dapat berfungsi sebagai bukti tertulis (otentik) dalam bentuk dokumen dimana isinya dapat dipercaya dan dipertanggungjawabkan.
- Sebagai pedoman kerja dalam melakukan suatu kegiatan atau aktivitas dimana surat resmi tersebut berisi tentang langkah-langkah kerja untuk pekerjaan tertentu.
- Sebagai alat untuk pengingat bagi si penerima surat baik itu perorangan, organisasi, atau lembaga. Itu sebabnya surat resmi perlu didokumentasikan agar dapat digunakan sebagai data di kemudian hari.
- Surat resmi juga berfungsi sebagai bukti historis dan bukti kronologis.

Meskipun saat ini telah ada fasilitas teknologi seperti email (surat elektronik) dan peran surat resmi dalam bentuk print out masih tetap diperlukan untuk keperluan tertentu, baik itu keperluan bisnis, niaga, pemberitahuan, dan keperluan



## 2. Ciri-Ciri Surat Resmi

Dapat diketahui bahwa untuk mengenali sebuah surat resmi bisa dilihat dari karakteristiknya. Berikut ini adalah ciri-ciri surat resmi sehingga dapat dikenali dengan mudah:

- Surat resmi menggunakan bahasa baku sesuai dengan kaidah bahasa Indonesia dan Ejaan Yang Disempurnakan (EYD) mulai dari kosa kata, frasa, hingga tata bahasa yang dipakai.
- Surat resmi dibuat dengan bahasa yang singkat, padat, dan efektif, serta mudah dimengerti maksud dan isinya.
- Surat resmi tidak menggunakan bahasa implisit, tapi bahasa eksplisit.
- Dibuat dalam bentuk full block atau semi block/ intended block
- Terdapat kop surat yang menyebutkan pihak yang mengeluarkan surat resmi tersebut.
- Didalam surat resmi selalu dicantumkan nomor surat, perihal, tanggal, alamat tujuan, dan lampiran bila ada.
- Surat resmi membutuhkan stempel atau cap khusus untuk kondisi tertentu.
- Surat resmi bentuknya sistematis dan dibuat sesuai aturan yang baku.

## 3. Jenis-Jenis Surat Resmi

Setelah memahami pengertian surat resmi, fungsi dan ciri-cirinya maka perlu diketahui jenis-jenis surat resmi. Berikut ini adalah macam-macam surat resmi:

### a) Surat Keputusan

Surat resmi ini bertujuan untuk menyampaikan keputusan dari atasan mengenai hal-hal yang selama ini belum jelas. Umumnya surat keputusan berhubungan dengan suatu instansi atau lembaga. Misalnya surat keputusan pengangkatan pegawai.

### b) Surat Permohonan

Surat permohonan digunakan ketika suatu pihak menyampaikan suatu permohonan kepada pihak lain. Misalnya surat permohonan bantuan dana, surat permohonan perceraian, dan lain-lain.



c) Surat Perintah

Surat resmi yang isinya memberikan instruksi kepada bawahan. Misalnya surat perintah perjalanan dinas, surat perintah untuk lembur, dan lain-lain.

d) Surat Kuasa

Surat resmi yang isinya memberikan kuasa atau wewenang kepada pihak lain. Misalnya surat kuasa untuk pembayaran pajak, surat kuasa pengambilan uang di bank, dan lain-lain.

e) Surat Panggilan/ Undangan

Surat resmi yang digunakan untuk memanggil atau mengundang seseorang untuk keperluan tertentu. Misalnya surat panggilan kerja, surat panggilan kepolisian, surat undangan pernikahan, dan lain-lain.

f) Surat Edaran

Surat resmi yang dibuat dan ditujukan kepada kalangan tertentu dimana isinya pemberitahuan kegiatan atau hal tertentu. Misalnya surat edaran pengumuman libur ujian.

#### 4. Struktur Surat Resmi

Seperti yang disebutkan pada pengertian surat resmi di atas, dalam surat resmi terdapat beberapa bagian penting yang harus dilengkapi. Beberapa bagian penulisan surat resmi tersebut adalah (Maxmanroe, 2018):

a) Kepala surat terdiri dari:

- Logo instansi/ Lembaga
- Nama instansi/ Lembaga
- Alamat, No. Telephone/ fax, dan email instansi/ lembaga

b) Nomor Surat

Setiap kali suatu instansi/ lembaga mengirimkan surat, pasti dilengkapi dengan nomor surat. Dengan adanya nomor surat tersebut, maka akan mudah mengetahui berapa jumlah surat yang telah dikeluarkan dalam satu bulan.



c) Tanggal Surat

Surat resmi selalu dilengkapi dengan tanggal pembuatan surat. Ini berguna sebagai informasi waktu dibuatnya surat tersebut.

d) Lampiran atau Hal

Terkadang surat resmi dilengkapi dengan lampiran, yaitu dokumen lain sebagai pendukung surat resmi tersebut.

e) Alamat Tujuan

Alamat tujuan diberikannya surat resmi tersebut. Biasanya penulisannya singkat karena alamat lengkap tujuan biasanya dicantumkan pada sampul surat.

f) Salam Pembuka

Kata pembuka dari sebuah surat resmi dimana bentuknya baku dan formal dengan bahasa yang sopan.

g) Isi Surat Tembusan

Tembusan dapat dibuat adalah bagian utama dari surat resmi. Informasi yang dimuat dalam bagian isi harus dibuat singkat, padat, dan jelas dengan menggunakan bahasa baku.

h) Salam Penutup

Selain salam pembuka, tentunya ada salam penutup surat. Hal ini untuk menunjukkan kesopanan dalam berkomunikasi melalui surat resmi.

i) Tanda Tangan Pengirim Surat

Pada bagian ini harus dicantumkan nama dan tanda tangan pengirim surat atau penanggungjawab. bila surat resmi tersebut perlu diketahui oleh pihak lain.



### 2.1.2 Desain Template

Sebuah dokumen atau file yang memiliki format preset, digunakan sebagai titik awal untuk aplikasi tertentu sehingga format tidak harus diciptakan kembali setiap kali digunakan (Rendi, Hari, & Anggraini, 2017).

Template dapat diartikan sebagai sebuah alat bantu yang berfungsi untuk memastikan bahwa semua dokumen memiliki layout yang standar. Secara sederhana, keberadaan template memungkinkan kita untuk menghasilkan dokumen yang seragam secara cepat (Didik Dwi Prasetya, 2016).

### 2.1.3 Web Aplikasi

Pada awalnya aplikasi web dibangun dengan hanya menggunakan bahasa yang disebut HTML (*Hyper Text Markup Language*). Pada perkembangan berikutnya, sejumlah skrip dan objek dikembangkan untuk memperluas kemampuan HTML. Web Aplikasi adalah adalah suatu aplikasi yang diakses menggunakan penjelajah web melalui suatu jaringan seperti Internet atau intranet. Ia juga merupakan suatu aplikasi perangkat lunak komputer yang dikodekan dalam bahasa yang didukung penjelajah web (seperti ASP, Perl, Java, Java Script, PHP, Python, Ruby, dll) dan bergantung pada penjelajah tersebut untuk menampilkan aplikasi.

Aplikasi web menjadi populer karena kemudahan tersedianya aplikasi klien untuk mengaksesnya, penjelajah web, yang kadang disebut sebagai suatu *thin client* (klien tipis). Kemampuan untuk memperbarui dan memelihara aplikasi web tanpa harus mendistribusikan dan menginstalasi perangkat lunak pada kemungkinan ribuan komputer klien merupakan alasan kunci popularitasnya. Aplikasi web yang umum misalnya webmail, toko ritel, lelang online, wiki, papan diskusi, weblog (Saiful, Ayu, & Abdi, 2013).

Keunggulan aplikasi berbasis web:

1. Dapat dijalankan dimanapun dan kapanpun tanpa harus melakukan instalasi



2. Tidak memerlukan sebuah lisensi ketika menggunakan aplikasi berbasis web dikarenakan lisensi merupakan sebuah tanggung jawab dari penyedia pelayanan aplikasi berbasis web
3. Dapat dijalankan atau digunakan di berbagai jenis sistem operasi, asalkan terhubung dengan jaringan internet
4. Dapat diakses melalui banyak media seperti komputer, laptop, notebook, handphone pintar yang sudah sesuai dengan WAP yang standart
5. Tidak memerlukan spesifikasi yang tinggi untuk melakukan dan menggunakan aplikasi jenis web ini.

Kekurangan aplikasi berbasis web:

1. Dibutuhkan sebuah jaringan atau koneksi internet untuk melakukan atau menggunakan aplikasi ini.
2. Dibutuhkannya sebuah keamanan yang baik hal ini dikarenakan pada jenis aplikasi berbasis web ini dijalankan secara terpusat, sehingga apabila sistem pusat mengalami down maka sistem aplikasi tidak bisa beroperasi dengan normal.

Keunggulan aplikasi berbasis dekstop:

1. Dapat berjalan dengan independen tanpa perlu menggunakan sebuah browser untuk mengaksesnya
2. Tidak memerlukan sebuah koneksi internet, hal ini disebabkan karena semua file yang diperlukan untuk menjalankan sebuah aplikasi ini sudah terinstal sebelumnya.
3. Proses lebih cepat
4. Mudah dalam pengaturan atau modifikasi

Kekurangan aplikasi berbasis dekstop:

1. Harus melakukan proses penginstalan aplikasi terlebih dahulu untuk bisa melakukan penggunaannya
2. Membutuhkan banyak lisensi pada setiap komputer
3. Aplikasi tidak bisa diakses dimanapun dan kapanpun.

memerlukan hardware dengan spesifikasi tinggi.

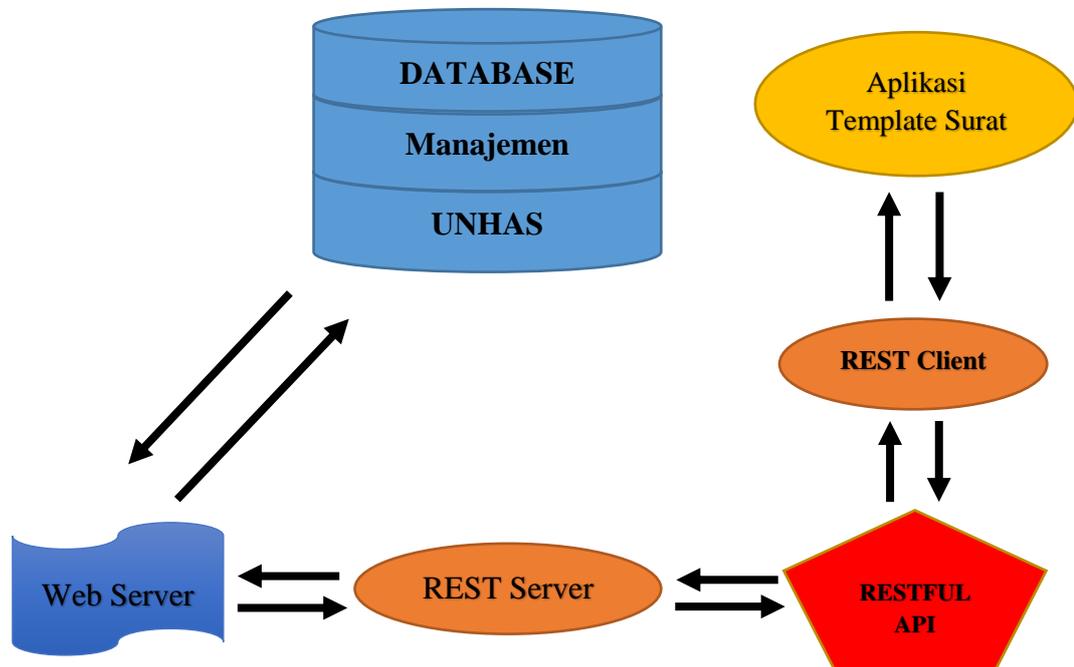


2.1.4 Restful API

RESTful API merupakan implementasi dari API (*Application Programming Interface*). REST (*Representational State Transfer*) adalah suatu arsitektur metode komunikasi yang menggunakan protokol HTTP untuk pertukaran data dan metode ini sering diterapkan dalam pengembangan aplikasi. Dimana tujuannya adalah untuk menjadikan sistem yang memiliki performa yang baik, cepat dan mudah untuk di kembangkan (*scale*) terutama dalam pertukaran dan komunikasi data (Irsan, 2018).

- **Cara kerja Restful API**

Pertama dari aplikasi template dilakukan *request* melalui REST API untuk mengambil data dari *database* manajemen persuratan UNHAS lalu dilakukan *HTTP Response* dengan memberikan data yang sesuai dengan apa yang *direquest* sebelumnya melalui *REST API*. Hal ini dilakukan melalui sebuah global ID atau URIs agar server dapat merespon dengan mengirimkan balik sebuah HTTP response sesuai yang di inginkan client.



Gambar 1. Cara kerja Restful API



*RESTFUL API* memiliki 4 komponen penting diantaranya adalah (Alam, Heni, & Rizal, 2018):

- *URL DESIGN*

*RESTful API* diakses menggunakan protokol HTTP. Penamaan dan struktur URL yang konsisten akan menghasilkan API yang baik dan mudah untuk dimengerti developer. URL API biasa disebut *endpoint* dalam pemanggilannya.

- *HTTP Verbs*

Setiap request yang dilakukan terdapat metode yang dipakai agar server mengerti apa yang sedang di *request client*

- a) GET

GET adalah metode HTTP *Request* yang paling mudah, metode ini digunakan untuk membaca atau mendapatkan data dari sumber.

- b) POST

POST adalah metode HTTP *Request* yang digunakan untuk membuat data baru dengan menyisipkan data dalam *body* saat *request* dilakukan.

- c) PUT

PUT adalah metode HTTP *Request* yang biasanya digunakan untuk melakukan *update* data *resource*.

- d) DELETE

DELETE adalah metode HTTP *Request* yang digunakan untuk menghapus suatu data pada *resource*.

- *HTTP Response Code*

HTTP *response code* adalah kode standarisasi dalam menginformasikan hasil request kepada *client*. Secara umum terdapat 3 kelompok yang biasa kita jumpai pada RESTful API yaitu:

- a. 2XX: adalah response code yang menampilkan bahwa request berhasil.

- b. 4XX: adalah response code yang menampilkan bahwa request mengalami kesalahan pada sisi client.



c. 5XX: adalah response code yang menampilkan bahwa request mengalami kesalahan pada sisi server.

- *Format Response*

Setiap *request* yang dilakukan *client* akan menerima data *response* dari server, response tersebut biasanya berupa data XML ataupun JSON. Setelah mendapatkan data response tersebut dan client bisa menggunakannya dengan cara memarsing data tersebut dan diolah sesuai kebutuhan.

### 2.1.5 PHP

PHP adalah bahasa pemrograman yang sering disisipkan ke dalam HTML. PHP sendiri berasal dari kata *Hypertext Preprocessor*. Sejarah PHP pada awalnya merupakan kependekan dari *Personal Home Page* (Situs personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama *Form Interpreted* (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari web.

Bahasa pemrograman ini menggunakan sistem *server-side*. *Server-side programming* adalah jenis bahasa pemrograman yang nantinya *script/program* tersebut akan dijalankan/diproses oleh server. Kelebihannya adalah mudah digunakan, sederhana, dan mudah untuk dimengerti dan dipelajari. Semenjak PHP menjadi bahasa pemrograman yang open source, pengembang tidak perlu menunggu sampai dengan *update* terbaru rilis.

Pengguna PHP akan lebih baik jika menggunakan versi terbaru. Sehingga jika telah rilis versi terbaru pengguna harus menyesuaikan sistem tersebut dengan versi PHP yang terbaru. Meskipun harus menggunakan versi terbaru, biaya untuk maintenance dan web *development* sangat terjangkau.

Bahasa pemrograman PHP membantu pengguna untuk mengembangkan

berbasis web yang cukup kompleks, handal, dan cepat. Tergantung dari si bisnis, penggunaan hosting, tingkat pengalaman, kebutuhan aplikasi, embangan timeframe. Selain itu ada banyak PHP frameworks yang dapat



dipilih. Untuk membuat sistem berbasis web tidak harus menggunakan PHP. Namun karena fiturnya yang menarik dan dirasa memudahkan, maka kebanyakan pengembangan menggunakan bahasa pemrograman ini.

Sebelum ada PHP, pengembang kebanyakan menggunakan bahasa pemrograman HTML. Bahasa pemrograman ini sudah dapat membuat tampilan website, namun tampilan yang dihasilkan terlihat biasa . Biasa karena website masih bersifat statis atau tetap, tidak bisa menyesuaikan tampilan dengan kondisi pengguna.

Semenjak adanya PHP maka pembuatan website dapat dikembangkan dan diatur agar menjadi website yang dinamis. Dimana website dapat menyesuaikan tampilan sesuai dengan perangkat yang digunakan oleh pengguna. Tidak hanya itu, website juga dapat melakukan input output dengan database. PHP sendiri biasanya disisipkan ke dalam kode HTML dan dikombinasikan dengan bahasa pemrograman lain, seperti CSS (Rini Sovia, 2011).

### 2.1.6 *CodeIgniter*

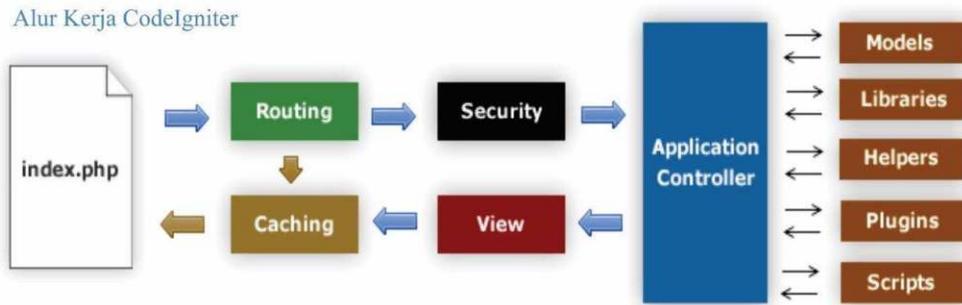
*Codeigniter* adalah sebuah framework untuk web yang dibuat dalam format PHP. Format yang dibuat ini selanjutnya dapat digunakan untuk membuat sistem aplikasi web yang kompleks. *Codeigniter* dapat mempercepat proses pembuatan web karena semua class dan modul yang dibutuhkan sudah ada dan programmer hanya tinggal menggunakannya kembali pada aplikasi web yang akan dibuat

Berikut ini adalah fitur-fitur penting yang ada di *CodeIgniter* (Dina, 2018):

- Sistem berbasis *Model-View-Controller* (MVC)
- Sangat Ringan dan Butuh Sedikit Sumber Daya (Memori + CPU)
- Mensupport berbagai macam database dan kaya fitur
- Mendukung *Query Builder* untuk mengakses database
- Validasi Form dan Data

mankan website dari XSS (*Cross Site Scripting*)  
men Session





Gambar 2. Alur Kerja Framework CodeIgniter

- *Index.php*: *Index.php* berfungsi sebagai file pertama dalam program yang akan dibaca oleh program.
- *The Router*: Router akan memeriksa HTTP request untuk menentukan hal apa yang harus dilakukan oleh program.
- *Cache File*: Apabila dalam program sudah terdapat “*cache file*” maka file tersebut akan langsung dikirim ke browser. File cache inilah yang dapat membuat sebuah website dapat di buka dengan lebih cepat. Cache file dapat melewati proses yang sebenarnya harus dilakukan oleh program *codeigniter*.
- *Security*: Sebelum file controller di *load* keseluruhan, HTTP request dan data yang disubmit oleh *user* akan disaring terlebih dahulu melalui fasilitas security yang dimiliki oleh *codeigniter*.
- *Controller*: Controller akan membuka file model, core libraries, helper dan semua resources yang dibutuhkan dalam program tersebut.
- *View*: Hal yang terakhir akan dilakukan adalah membaca semua program yang ada dalam view file dan mengirimkannya ke browser supaya dapat dilihat. Apabila *file view* sudah ada yang di “*cache*” maka *file view* baru yang belum ter-*cache* akan mengupdate file *view* yang sudah ada (Prabowo, 2015).

### 2.1.7 My SQL



*MySQL* adalah sebuah perangkat lunak system manajemen basis data *SQL* (DBMS) yang multithread, dan multi-user. *MySQL* adalah implementasi dari sistem manajemen basisdata relasional (RDBMS). *MySQL* dibuuh oleh *TcX*

dan telah dipercaya mengelola system dengan 40 buah database berisi 10.000 tabel dan 500 di antaranya memiliki 7 juta baris.

Ada tiga bentuk *SQL* yang perlu diketahui, yaitu (Yasin, 2019):

- *Data Definition Language (DDL)*

DDL berguna pada saat pengguna ingin mendefinisikan data didalam database. Terdapat beberapa query yang dikelompokkan ke dalam DDL, yaitu: Argumen DDL di atas perlu dipahami karena merupakan dasar penggunaan *SQL* di bagian awal pembuatan database. Sebagai contoh jika belum menjalankan perintah “CREATE”, maka belum bisa melanjutkan penggunaan argumen yang lainnya.

- *Data Manipulation Language (DML)*

DML dapat dipakai setelah menjalankan perintah DDL. DML berfungsi untuk memanipulasi, mengubah, atau mengganti isi dari database (tabel) yang sudah ada.

Terdapat beberapa perintah DML yang perlu diketahui, yaitu:

*INSERT* : Dipakai untuk memasukkan data ke dalam tabel pada database.

*UPDATE* : Dipakai untuk mengubah data yang ada di dalam tabel pada database.

*DELETE* : Dipakai untuk menghapus data di dalam tabel pada database.

- *Data Control Language (DCL)*

Saat telah tersedia user dan ingin mengatur hak akses masing-masing user, maka sebaiknya memahami berbagai macam jenis DCL dan cara penggunaannya. DCL berguna untuk memberikan hak akses database, mendefinisikan space, mengalokasikan space, dan melakukan audit penggunaan database.



Terdapat beberapa perintah DCL yang perlu Anda ketahui, yaitu:

- GRANT* : Dipakai untuk memberikan izin kepada user untuk mengakses database.
- REVOKE* : Dipakai untuk membatalkan izin user untuk mengakses database.
- COMMIT* : Dipakai untuk menetapkan penyimpanan pada database.
- ROOLBACK* : Dipakai untuk membatalkan penyimpanan pada database.

Pada saat ini *MySQL* merupakan database server yang sangat terkenal di dunia, hal tersebut terjadi karena bahasa dasar yang digunakan untuk mengakses database yaitu *SQL*. *SQL (Structured Query Language)* pertama kali diterapkan pada sebuah proyek riset pada laboratorium riset San Jose, IBM yang bernama *system R*. Kemudian *SQL* juga dikembangkan oleh *Oracle*, *Informix* dan *Sybase*. Dengan menggunakan *SQL*, proses pengaksesan database lebih user-friendly dibandingkan dengan yang lain, misalnya *dBase* atau *Clipper* karena mereka masih menggunakan perintah-perintah pemrograman murni.

*SQL* statement menginstruksikan server untuk menjalankan operasi tertentu (Rini Sovia, 2011) :

- *Data query*: meminta informasi yang spesifik dari database yang sudah ada.
- Manipulasi data: menambahkan, menghapus, mengubah, menyortir, melakukan operasi lainnya untuk memodifikasi data, *value*, atau *visual*.
- Identitas data (*data identity*): menentukan tipe data, misalnya mengubah data numerik menjadi data integer. Selain itu, juga menentukan schema atau hubungan dari masing-masing tabel yang ada di database.
- *Data access control*: menyediakan metode keamanan untuk melindungi data, termasuk dalam menentukan siapa yang boleh melihat atau menggunakan informasi yang tersimpan di database.



2.2 Kerangka Pikir

