

**Perancangan dan Implementasi Sistem Pelayanan Puskesmas
Menggunakan *Framework* CodeIgniter Pada Puskesmas Kaliwungu
Kab. Semarang**

Artikel Ilmiah



Peneliti:

**Setya Angga Cahya (672015243)
Yeremia Alfa Susetyo, S.Kom, M.Cs.**

**Program Studi Teknik Informatika
Fakultas Teknologi Informasi
Universitas Kristen Satya Wacana
Salatiga
2019**



PERNYATAAN TIDAK PLAGIAT

Saya yang bertanda tangan di bawah ini:

Nama : SETYA ANGGA CAHYA
NIM : 672015243 Email : setyaanggac1998@gmail.com
Fakultas : TEKNOLOGI INFORMASI Program Studi : TEKNIK INFORMATIKA
Judul tugas akhir : PERANCANGAN DAN IMPLEMENTASI SISTEM PELAYANAN
PUSKESMAS MENGGUNAKAN FRAMEWORK CODEIGNITER PADA
PUSKESMAS KALIWUNGU KAB. SEMARANG
Pembimbing : 1. Yeremia Alfa Susetyo, S.Kom, M.Cs.
2. _____

Dengan ini menyatakan bahwa:

1. Hasil karya yang saya serahkan ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar kesarjanaan baik di Universitas Kristen Satya Wacana maupun di institusi pendidikan lainnya.
2. Hasil karya saya ini bukan saduran/terjemahan melainkan merupakan gagasan, rumusan, dan hasil pelaksanaan penelitian/implementasi saya sendiri, tanpa bantuan pihak lain, kecuali arahan pembimbing akademik dan narasumber penelitian.
3. Hasil karya saya ini merupakan hasil revisi terakhir setelah diujikan yang telah diketahui dan disetujui oleh pembimbing.
4. Dalam karya saya ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali yang digunakan sebagai acuan dalam naskah dengan menyebutkan nama pengarang dan dicantumkan dalam daftar pustaka.

Pernyataan ini saya buat dengan sesungguhnya. Apabila di kemudian hari terbukti ada penyimpangan dan ketidakbenaran dalam pernyataan ini maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya saya ini, serta sanksi lain yang sesuai dengan ketentuan yang berlaku di Universitas Kristen Satya Wacana.





PERNYATAAN PERSETUJUAN AKSES

Saya yang bertanda tangan di bawah ini:

Nama : SETYA ANGGA CAHYA
NIM : 692015242 Email : Setyanangga1998@gmail.com
Fakultas : TEKNOLOGI INFORMASI Program Studi : TEKNIK INFORMATIKA
Judul tugas akhir : PERANCANGAN DAN IMPLEMENTASI SISTEM PELAYANAN
PUSKESMAS MENGGUNAKAN FRAMEWORK COGNITIF PADA
PUSKESMAS KALIWUNGU KAB. SEMARANG

Dengan ini saya menyerahkan hak *non-eksklusif** kepada Perpustakaan Universitas – Universitas Kristen Satya Wacana untuk menyimpan, mengatur akses serta melakukan pengelolaan terhadap karya saya ini dengan mengacu pada ketentuan akses tugas akhir elektronik sebagai berikut (beri tanda pada kotak yang sesuai):

- a. Saya mengizinkan karya tersebut diunggah ke dalam aplikasi Repositori Perpustakaan Universitas, dan/atau portal GARUDA
- b. Saya tidak mengizinkan karya tersebut diunggah ke dalam aplikasi Repositori Perpustakaan Universitas, dan/atau portal GARUDA**

* Hak yang tidak terbatas hanya bagi satu pihak saja. Pengajar, peneliti, dan mahasiswa yang menyerahkan hak non-eksklusif kepada Repositori Perpustakaan Universitas saat mengumpulkan hasil karya mereka masih memiliki hak copyright atas karya tersebut.
** Hanya akan menampilkan halaman judul dan abstrak. Pilihan ini harus dilampiri dengan penjelasan alasan tertulis dari pembimbing TA dan diketahui oleh pimpinan fakultas (dekan/kaprodi).

Demikian pernyataan ini saya buat dengan sebenarnya.

Salatiga, 15 Januari 2020

Mengetahui,

Yeremia Alfa Fusetyo, S.Kom, M.Cs.

Tanda tangan & nama orang pembimbing I

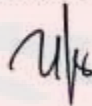
Setya Angga Cahya.

Tanda tangan & nama orang mahasiswa

Lembar Pengesahan

Judul Tugas Akhir : Perancangan dan Implementasi Sistem Pelayanan Puskesmas menggunakan Framework CodeIgniter pada Puskesmas Kaliwungu Kab. Semarang
Nama Mahasiswa : SETYA ANGGA CAHYA
NIM : 672015243
Program Studi : Teknik Informatika
Fakultas : Teknologi Informasi

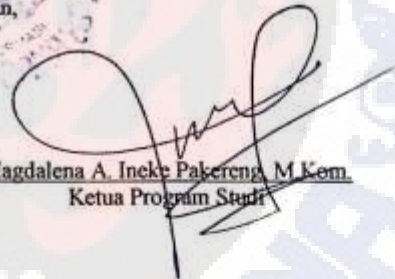
Menyetujui,



Yerima Alfa Susetyo, S.Kom., M.Cs.
Pembimbing



Dr. Wiwin Sulistyono, ST., M.Kom.
Dekan



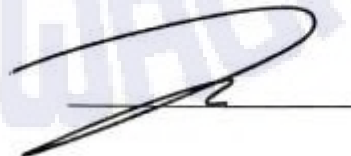
Magdalena A. Ineke Pakere, M.Kom.
Ketua Program Studi

1956

Dinyatakan Lulus Tanggal: 8 Januari 2020

Reviewer :

- Radius Tanone, S.Kom., M.Cs.



**Perancangan dan Implementasi Sistem Pelayanan Puskesmas
menggunakan *Framework CodeIgniter* pada Puskesmas Kaliwungu Kab.
Semarang**

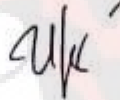
Oleh

**SETYA ANGGA CAHYA
672015243**

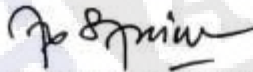
LAPORAN PENELITIAN

Diajukan Kepada Program Studi Teknik Informatika Guna Memenuhi Sebagian Dari
Persyaratan Untuk Mencapai Gelar Sarjana Komputer

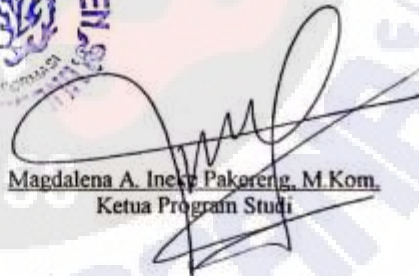
Disetujui oleh,



Yerima Alfa Susetyo, S.Kom., M.Cs.
Pembimbing



Dr. Wiwin Sulistyono, ST., M.Kom.
Dekan



Magdalena A. Ines Pakoreng, M.Kom.
Ketua Program Studi

**FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN SATYA WACANA
SALATIGA
2020**

**Perancangan dan Implementasi Sistem Pelayanan Puskesmas
menggunakan Framework CodeIgniter pada Puskesmas Kaliwungu Kab.
Semarang**

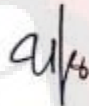
Oleh

SETYA ANGGA CAHYA
672015243

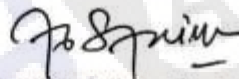
ARTIKEL ILMIAH

Diajukan Kepada Program Studi Teknik Informatika Guna Memenuhi Sebagian Dari
Persyaratan Untuk Mencapai Gelar Sarjana Komputer

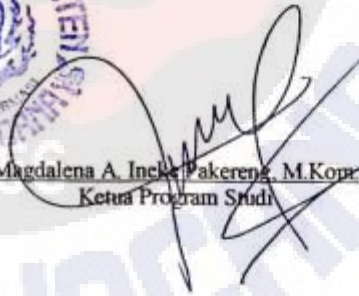
Disetujui oleh,



Yerima Alfa Susetyo, S.Kom., M.Cs.
Pembimbing



Dr. Wiwin Sulistyono, ST., M.Kom.
Dekan



Magdalena A. Inek, M.Kom.
Ketua Program Studi

**FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN SATYA WACANA
SALATIGA
2020**

1. Pendahuluan

Pusat Kesehatan Masyarakat (Puskesmas) adalah fasilitas pelayanan kesehatan yang menyelenggarakan upaya kesehatan masyarakat dan upaya kesehatan perseorangan tingkat pertama, dengan lebih mengutamakan upaya promotif dan preventif, untuk mencapai derajat kesehatan masyarakat yang setinggi-tingginya di wilayah kerjanya. Puskesmas mempunyai tugas melaksanakan kebijakan kesehatan untuk mencapai tujuan pembangunan kesehatan di wilayah kerjanya dalam rangka mendukung terwujudnya kecamatan sehat [1].

Berdasarkan hasil pengamatan dan wawancara dengan Kepala Puskesmas Kaliwungu. Didapatkan hasil bahwa proses pencatatan rekam medis di Puskesmas Kaliwungu sudah menggunakan aplikasi dari pemerintah yang merupakan sistem informasi pelayanan pasien yang ditujukan untuk pasien peserta BPJS (Badan Penyelenggara Jaminan Sosial) secara *online*. Namun, kendala yang ditemukan apabila menggunakan sistem informasi dari BPJS adalah sistem informasi ini hanya dapat digunakan untuk pasien yang menjadi peserta BPJS saja. Sedangkan saat ini masih ditemukan pasien umum yang bukan merupakan peserta BPJS, sehingga pencatatan rekam medis masih menggunakan pencatatan manual menggunakan kertas.

Kondisi lain yang terjadi saat ini, pengolahan data pasien anggota puskesmas juga masih menggunakan pencatatan manual menggunakan kertas. Sehingga saat pasien datang ke puskesmas untuk berobat, petugas harus memastikan dahulu bahwa pasien tersebut apakah sudah tercatat sebagai anggota puskesmas atau belum. Petugas mendapatkan kesulitan mencari data pasien tersebut, sebab pencatatan masih ditulis manual dalam buku. Terlebih lagi, harus membuka setiap berkas apabila pencarian tersebut tidak menemukan nama yang dicari sehingga membutuhkan waktu cukup lama. Dengan pencatatan data anggota puskesmas dan rekam medis pasien yang masih menggunakan kertas, keadaan seperti ini bisa berakibat hilangnya berkas rekam medis pasien. Dari kendala-kendala yang terjadi, dapat mengakibatkan pelayanan kesehatan kepada pasien terganggu. Sehingga, aktifitas yang dilakukan dapat menyebabkan waktu yang lama dan tidak efisien..

CodeIgniter adalah *framework* aplikasi berbasis *web* yang mengusung model MVC (*Model, View, Controller*). Salah satu *framework* PHP ini dapat menjadi *tools* bagi seorang *web developer* untuk mengembangkan suatu situs dengan lebih mudah. CodeIgniter bersifat *open source* (tidak berbayar) sehingga *framework* PHP ini dapat dengan mudah didapatkan. Keunggulan CodeIgniter adalah performa yang cepat, konfigurasi sederhana, banyak komunitas, dan dokumentasi yang lengkap [2].

Berdasarkan latar belakang yang telah dikemukakan, maka rumusan masalah yang dapat diambil dalam penelitian ini adalah bagaimana membuat sistem informasi berbasis *web* yang dapat membantu proses pelayanan kesehatan pada Puskesmas Kecamatan Kaliwungu. Oleh karena itu dalam penelitian ini akan membangun suatu sistem informasi layanan kesehatan berbasis *web* menggunakan *framework* CodeIgniter yang berisi aplikasi dapat mempermudah kinerja dari puskesmas dalam pengolahan data.

Tujuan penelitian ini untuk merancang Sistem Pelayanan Puskesmas menggunakan *Framework* CodeIgniter pada Puskesmas Kaliwungu. Batasan masalah dalam penelitian ini adalah Aplikasi ini dibuat berbasis *web* menggunakan bahasa pemrograman PHP dengan *framework* CodeIgniter. Aplikasi ini berisi informasi kegiatan pelayanan kesehatan di Puskesmas Kecamatan Kaliwungu yang meliputi data petugas, data pasien, data dokter, dan data pelayanan pengobatan. Aplikasi ini hanya menangani proses rawat jalan.

2. Tinjauan Pustaka

Perancangan Sistem Informasi Layanan Kesehatan Puskesmas Ngemplak Kabupaten Boyolali, aplikasi ini dirancang untuk membantu proses pengolahan data yang masih secara manual di puskesmas dapat diganti menggunakan komputer. Aplikasi yang dibangun berbasis *web* menggunakan bahasa pemrograman PHP dan menggunakan *database* MySQL. Hasilnya sistem informasi layanan kesehatan pada Puskesmas Ngemplak Boyolali berbasis *web* menghasilkan informasi seputar layanan kesehatan dan pengolahan data pasien secara lebih lengkap, jelas, dan mudah diakses. Sehingga mempermudah dalam pengolahan data pasien yang ada di Puskesmas Ngemplak Boyolali [3]. Berbeda dengan penelitian sebelumnya yang dalam pengembangan sistem menggunakan PHP *Native*. Pada penelitian ini menggunakan *framework* CodeIgniter dalam pengembangan sistem pelayanan untuk Puskesmas Kaliwungu. Kelebihan menggunakan *framework* CodeIgniter dibanding menggunakan PHP *Native* adalah dapat mempercepat dan mempermudah pembangunan sebuah aplikasi *web*, relatif memudahkan dalam proses *maintenance* karena sudah ada pola tertentu dalam sebuah *framework* (dengan syarat programmer mengikuti pola standar yang ada), umumnya *framework* menyediakan fasilitas-fasilitas yang umum dipakai sehingga kita tidak perlu membangun dari awal (misalnya validasi, ORM, *pagination*, *multiple database*, *scaffolding*, pengaturan *session*, *error handling*, dll) [8].

Sistem Informasi Layanan Kesehatan Dengan Menggunakan Codeigniter Pada Puskesmas Bululawang, proses pelayanan pada Puskesmas Bululawang masih menggunakan pencatatan manual dalam melayani pasien. Sehingga dilakukan pembuatan sistem informasi layanan kesehatan berbasis *web* dengan menggunakan bahasa pemrograman PHP dengan *framework* Codeigniter. Hasilnya Pencatatan dan akses data pasien, pengolahan data rekam medis, dan akses riwayat rekam medis pasien dapat diakses dengan mudah oleh petugas kesehatan. Aplikasi sistem informasi Puskesmas dapat meningkatkan kualitas pelayanan kesehatan masyarakat oleh petugas kesehatan kepada pasien [4]. Pada penelitian terdahulu hanya terdapat tiga *actor* dalam sistem yang dibuat, yakni Administrator, Petugas Loker dan Petugas Kesehatan. Pada penelitian ini terdapat empat *actor*, *Administrator*, Petugas Loker (Petugas Pendaftaran), Petugas Kesehatan dan Petugas Apotek. Perbedaan terdapat pada Petugas Apotek, dimana Petugas Apotek ini hanya khusus mengelola data obat-obatan beserta stoknya.

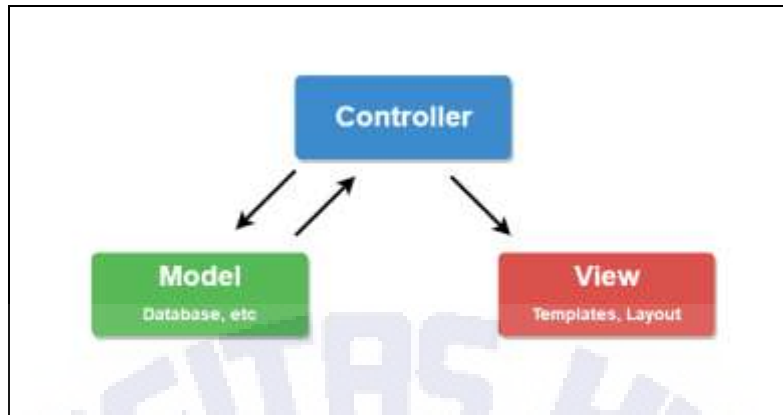
Perancangan Sistem Informasi Rawat Jalan Berbasis *Web* Pada Puskesmas Winong, dengan latar belakang untuk menginginkan akses terhadap pelayanan kesehatan yang berkualitas dapat ditingkatkan melalui peningkatan kinerja Puskesmas. Sehingga membuat sistem informasi berbasis *web* dengan bahasa pemrograman yang digunakan antara lain PHP, ASP.Net dan memanfaatkan *database* MySQL. Hasilnya dengan menggunakan sistem informasi rawat jalan ini, dapat mempercepat pelayanan kesehatan pada Puskesmas Winong [5]. Pada penelitian sebelumnya yang dalam pengembangan sistem menggunakan *database* MySQL. Pada penelitian ini menggunakan *database* MariaDB dalam pengembangan sistem pelayanan untuk Puskesmas. MariaDB dirancang sebagai pengganti MySQL yang pada awalnya dirilis pada Januari 2009. Para pengembang MariaDB terdiri dari mantan pengembang MySQL yang asli. Berdasarkan pengujian, apabila dilihat dari sudut pandang kecepatan *query*, pemenangnya adalah MariaDB. Dalam pengujian data *query* yang diurutkan berdasarkan *GeoLocation*, MariaDB berhasil mengungguli MySQL dengan margin yang cukup besar [11].

Berdasarkan ketiga penelitian yang sudah dilakukan sebelumnya, terdapat keterkaitan dengan penelitian yang akan dilakukan. Proses pengolahan data pada puskesmas yang masih dilakukan secara manual. Pemanfaatan temuan dari ketiga penelitian terdahulu yang berguna bagi penelitian ini adalah pada penggunaan bahasa pemrograman yang sama yaitu berbasis PHP dan penggunaan *database* yang sama MySQL. Dari penelitian-penelitian yang telah disebutkan, dan dengan ditemukannya fakta bahwa *framework* CodeIgniter memiliki keunggulan yang mampu mempercepat dan mempermudah pembangunan sebuah aplikasi *web*. Serta keunggulan MariaDB yang menawarkan performansi yang lebih unggul dari MySQL. Maka penelitian ini bertujuan untuk membuat Sistem Pelayanan Puskesmas Berbasis Web Menggunakan *Framework* CodeIgniter Pada Puskesmas Kaliwungu.

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan [6].

Bahasa Pemrograman PHP adalah bahasa pemrograman *script server-side* yang didesain untuk pengembangan *web*. Selain itu, PHP juga bisa digunakan sebagai bahasa pemrograman umum (wikipedia). Pertama kali dikembangkan oleh Rasmus Lerdorf pada tahun 1995, dan sekarang PHP dikembangkan oleh The PHP Group. Pada awalnya PHP dikenal dengan singkatan *Personal Home Page*. Karena *server* tersebut diperuntukan untuk *website* pribadi. Tetapi untuk saat ini PHP sudah bermetamorfosis menjadi bahasa pemrograman yang sangat populer yang digunakan untuk *website* terkenal seperti Wikipedia, WordPress, Joomla dll. Untuk saat ini PHP dikenal dengan singkatan Hypertext Preprocessor sebuah kepanjangan rekursif, yakni permainan kata dimana kepanjangannya terdiri dari singkatan itu sendiri. Bahasa pemrograman PHP banyak digunakan karena sifatnya yang *open source* yaitu dapat digunakan secara gratis [7].

CodeIgniter adalah sebuah *web application framework* yang bersifat *open source* yang digunakan untuk membangun aplikasi PHP dinamis. CodeIgniter menjadi sebuah *framework* PHP dengan model MVC (*Model, View, Controller*) untuk membangun *website* dinamis dengan menggunakan PHP yang dapat mempercepat pengembang untuk membuat sebuah aplikasi *web*. Selain ringan dan cepat, CodeIgniter juga memiliki dokumentasi yang super lengkap disertai dengan contoh implementasi kodenya. Dokumentasi yang lengkap inilah yang menjadi salah satu alasan kuat mengapa banyak orang memilih CodeIgniter sebagai *framework* pilihannya [8]. CodeIgniter didasarkan pada pola pengembangan *Model-View-Controller*. MVC adalah pendekatan perangkat lunak yang memisahkan logika aplikasi dari presentasi. Dalam praktiknya, ini memungkinkan halaman *web* Anda untuk mengandung skrip minimal karena presentasi terpisah dari skrip PHP.



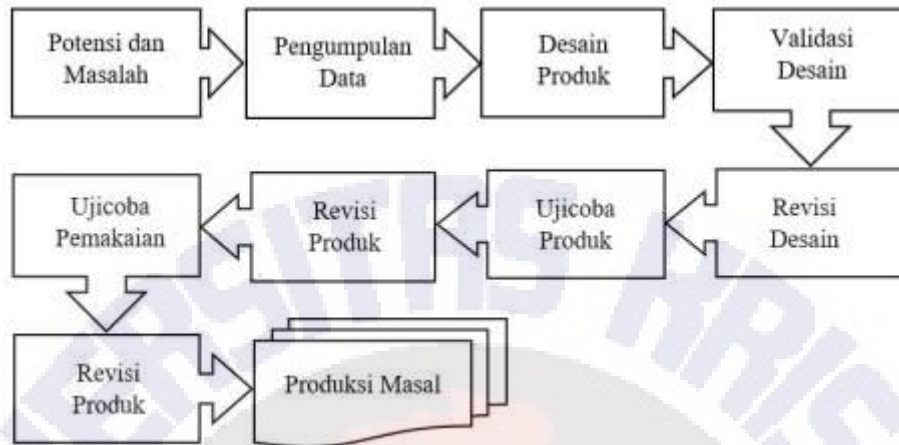
Gambar 1 Bagan MVC

- *Model* mewakili struktur data. Biasanya kelas *model* akan berisi fungsi yang membantu mengambil, menyisipkan, dan memperbarui informasi dalam basis data.
- *View* adalah informasi yang disajikan kepada pengguna. Tampilan biasanya akan menjadi halaman *web*, tetapi dalam CodeIgniter, tampilan juga bisa berupa *fragmen* halaman seperti *header* atau *footer*. Ini juga bisa berupa halaman RSS, atau "halaman" jenis lainnya.
- *Controller* berfungsi sebagai perantara antara *Model*, *View*, dan sumber daya lainnya yang diperlukan untuk memproses permintaan HTTP dan menghasilkan halaman *web* [13].

MariaDB dirancang sebagai pengganti MySQL yang pada awalnya dirilis pada Januari 2009. Para pengembang MariaDB terdiri dari mantan pengembang MySQL yang asli. Khawatir perubahan dimasa yang akan datang setelah akuisisi Sun Microsystems oleh Oracle, dan dengan ekstensi properti intelektual MySQL, tim pemrograman asli MySQL keluar untuk mendorong pembuatan percabangan dari MySQL berbasis komunitas bernama MariaDB. Oleh karena dibangun berdasarkan perintah dasar MySQL, *libraries*, dan *database* akan kompatibel antara *database* relasional MySQL dan MariaDB. Fitur API juga dibawa dari MySQL. Fitur utama MariaDB adalah sama dengan fitur dari MySQL dengan kemampuan tambahan untuk sekadar mengganti MySQL disebagian besar lingkungan dimana ia digunakan. Karena MariaDB sangat mirip, sebagian besar aplikasi untuk MySQL berfungsi dengan MariaDB tanpa modifikasi apa pun [9]. MariaDB juga mencakup fitur *Geographic Information Systems* (GIS) dan *JavaScript Object Notation* (JSON). Package MSI yang saat ini untuk MariaDB juga berisi HeidiSQL yang merupakan klien grafis *open source* yang dapat digunakan untuk berinteraksi dengan MariaDB. Pengembang dari MariaDB bekerja dengan orang-orang HeidiSQL untuk memastikan dapat mendukung semua fitur dan opsi di MariaDB [10].

3. Perancangan Sistem

Perancangan Sistem yang akan dilakukan dalam Sistem Pelayanan Kesehatan pada Puskesmas Kaliwungu menggunakan *framework* CodeIgniter dapat dilihat pada Gambar 2.



Gambar 2 Metodologi Penelitian *Research and Development* [12]

Tahap pertama merupakan tahap Potensi dan Masalah. Pada tahap ini ditentukan potensi dan masalah apa yang menjadi dasar dalam melakukan penelitian serta pembuatan sistem pelayanan Puskesmas Kaliwungu. Potensi dan masalah yang ditentukan harus berdasarkan fakta dan merupakan suatu hal yang perlu untuk segera ditemukan solusinya.

Tahap selanjutnya adalah Pengumpulan Data. Dilakukan pengumpulan informasi yang berkaitan dengan masalah dan potensi yang ada. Pengumpulan data dapat dilakukan dengan metode observasi dan studi pustaka. Informasi yang didapat berguna untuk merencanakan produk yang diharapkan mampu mengatasi permasalahan yang ada.

Desain produk dari aplikasi *web* yang akan dibuat ditentukan pada tahap selanjutnya, yaitu tahap Desain Produk. Desain produk dapat diwujudkan dalam bentuk gambar atau bagan, sehingga dapat digunakan sebagai pegangan untuk menilai dan membuatnya. Kemudian pada tahap Validasi Desain akan dinilai apakah rancangan produk secara rasional akan lebih efektif atau tidak. Dikatakan secara rasional, karena validasi ini masih bersifat penilaian berdasarkan pemikiran rasional, belum fakta lapangan.

Tahap berikutnya adalah Revisi Desain. Setelah desain produk divalidasi, maka akan diketahui kelemahan serta kekuatan dari desain produk tersebut. Kelemahan tersebut selanjutnya akan dicoba untuk dikurangi dengan cara memperbaiki desain produk tersebut, agar dapat menghasilkan desain produk yang lebih efektif. Desain produk yang telah dibuat tidak dapat langsung diujicoba, tetapi harus dibuat terlebih dahulu menjadi barang atau prototipe. Barang atau prototipe tersebut yang kemudian diujicoba pada tahap Uji coba Produk. Pada proses pengembangan sistem pelayanan Puskesmas Kaliwungu, tahap pengujian ini dilakukan oleh seluruh petugas kesehatan yang ada di Puskesmas Kaliwungu. Pengujian dilakukan dengan tujuan untuk mendapatkan informasi apakah produk tersebut sudah efektif dan efisien dalam mengatasi masalah. Berdasarkan hasil pengujian produk, jika masih terdapat kelemahan pada produk maka akan segera dilakukan perbaikan.

Setelah pengujian terhadap produk berhasil, maka produk yang berupa aplikasi *web* akan diujicoba dalam lingkup yang lebih luas. Dalam uji coba ini tetap harus dinilai kekurangan atau hambatan yang muncul guna perbaikan lebih lanjut. Selanjutnya, dilakukan revisi produk kembali apabila dalam pemakaian dalam kondisi nyata masih terdapat kekurangan atau

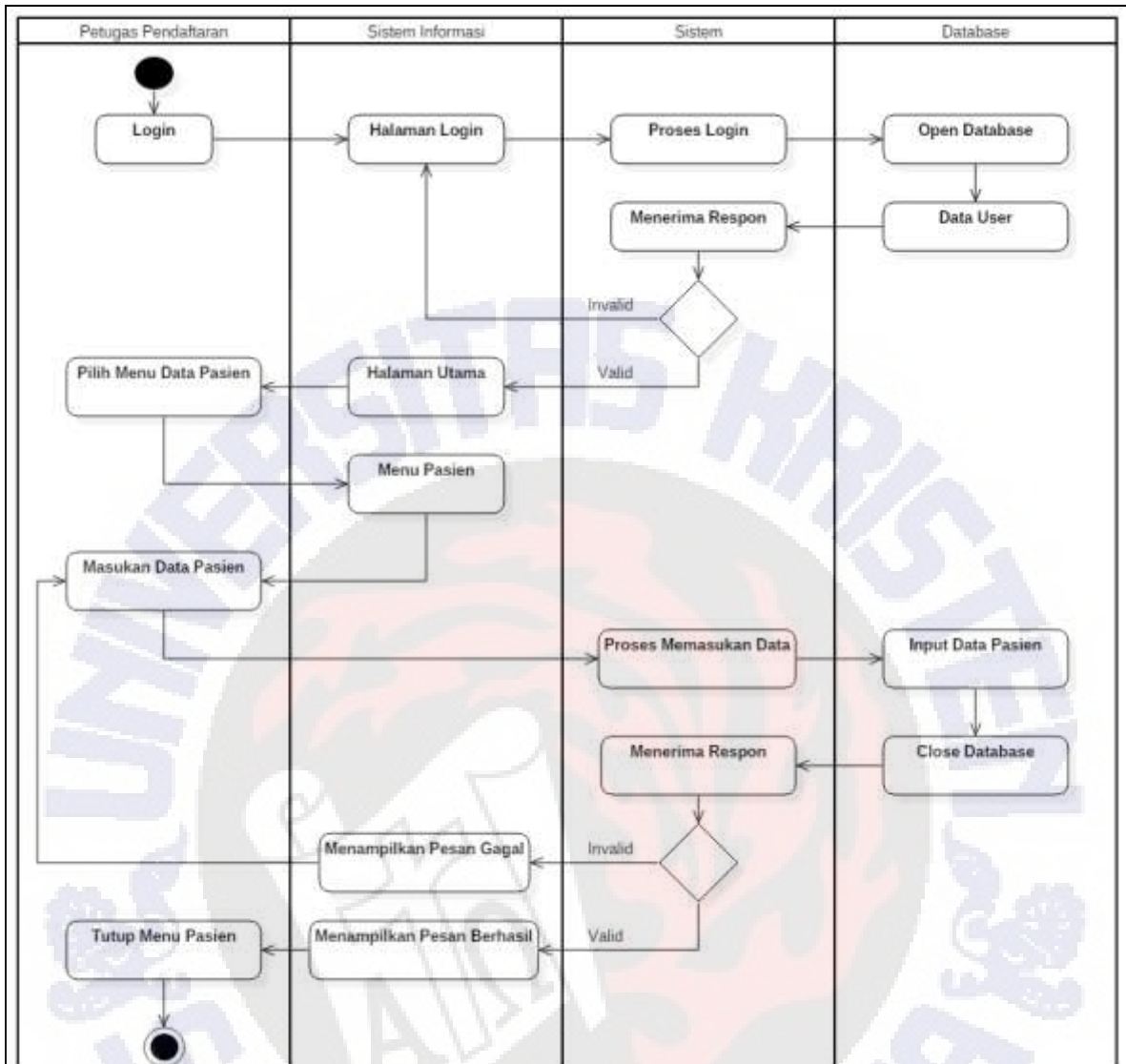
kelemahan. Produk akan dievaluasi untuk penyempurnaan produk. Tahap terakhir adalah Produksi Masal. Produksi masal dilakukan apabila produk, dalam penelitian ini merupakan aplikasi *web*, dinyatakan efektif dan layak untuk digunakan oleh pengguna. Produksi masal akan dilakukan dengan menerapkan sistem pelayanan ini untuk digunakan di Puskesmas Kaliwungu.

Perancangan sistem pelayanan Puskesmas Kaliwungu ini dibuat menggunakan *Unified Modelling Language (UML)* yang meliputi *Use Case Diagram*, *Activity Diagram* dan *Class Diagram*.



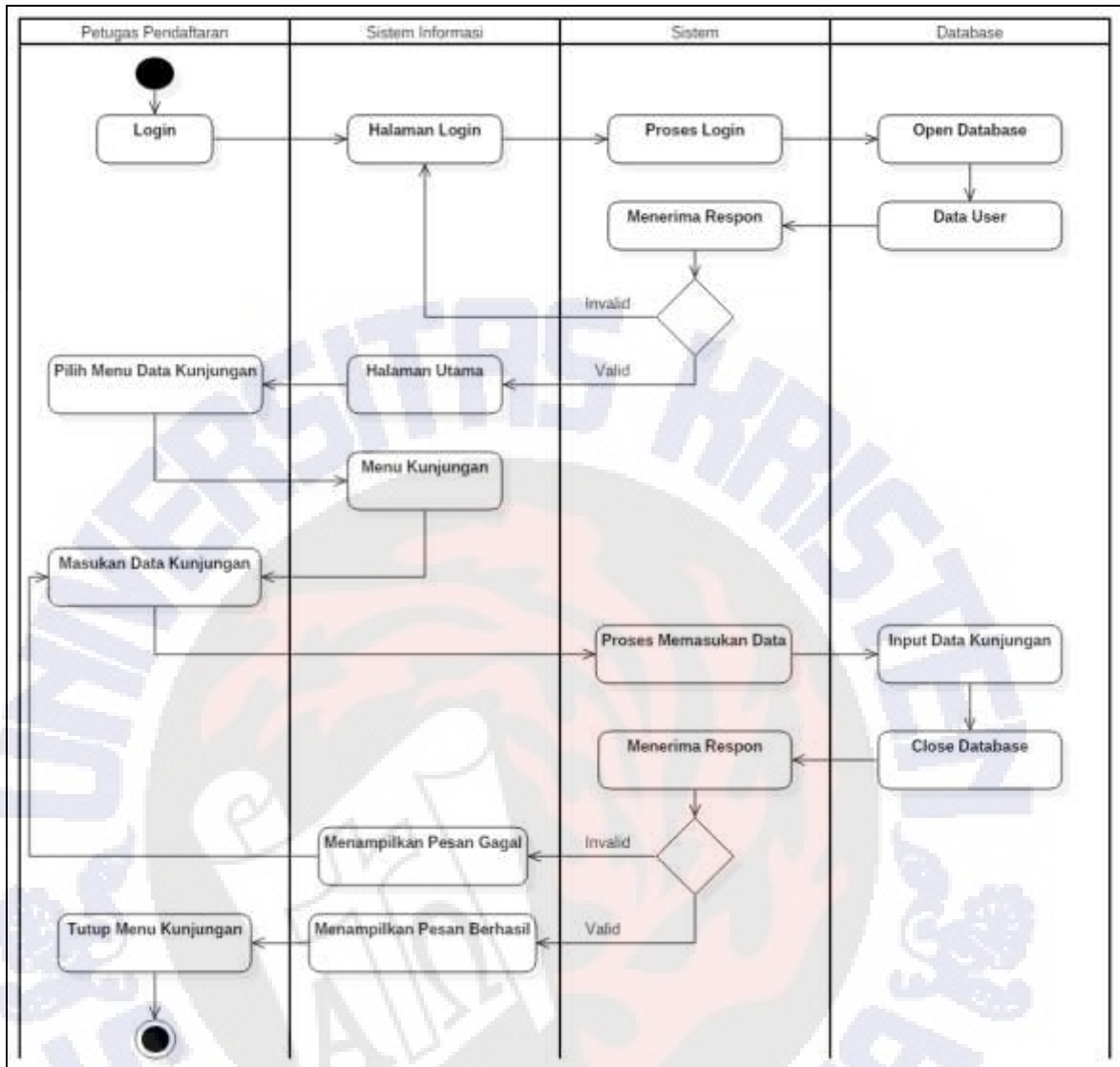
Gambar 3 Use Case Diagram

Gambar 3 menjelaskan bahwa terdapat empat aktor yang berperan pada aplikasi ini. Admin memiliki hak untuk mengakses dan mengelola data yang meliputi data jabatan, data pengguna, data level pengguna, data pegawai, data poli, dan data dokter. Admin hanya bisa melihat data diagnosa, data obat, data tindakan, data tindakan berobat, data detil tindakan berobat, data pasien, data berobat, dan data kunjungan. Sedangkan Petugas Pendaftaran dapat mengakses dan mengelola data pasien dan data kunjungan. Petugas Kesehatan memiliki hak untuk mengakses dan mengelola data yang meliputi data diagnosa, data tindakan, dan data tindakan berobat. Aktor terakhir, Petugas Apotek hanya memiliki hak untuk mengakses dan mengelola data obat.



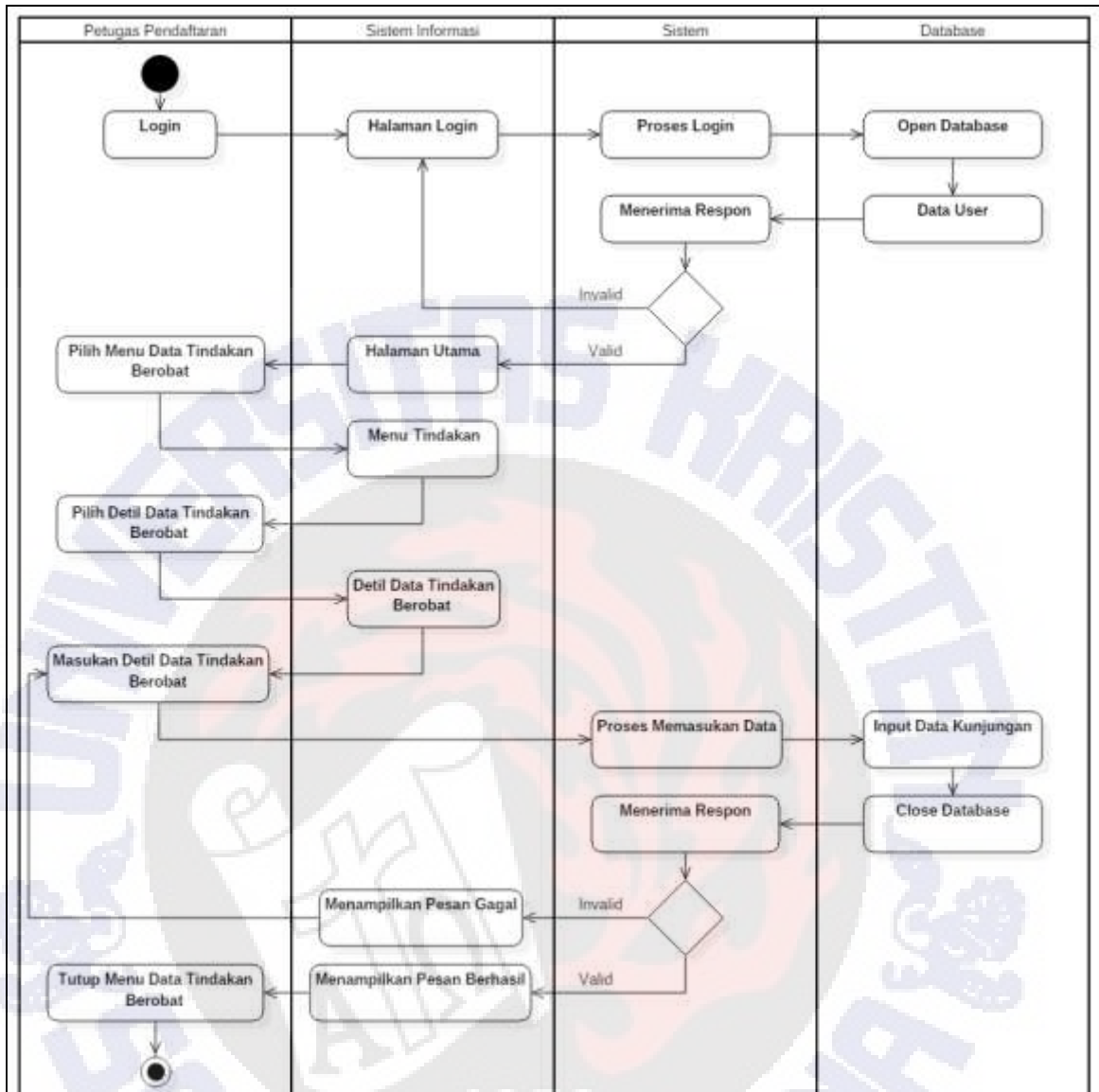
Gambar 4 Activity Diagram Menu Input Data Pasien

Gambar 4 menjelaskan tentang *activity diagram* pada menu Data Pasien. Aktivitas petugas pendaftaran untuk menginput data pasien sebagai anggota puskesmas diawali dengan proses *login* ke dalam aplikasi. Setelah *email* dan *password* sudah terverifikasi dan valid, maka akan ditampilkan halaman menu utama pada aplikasi. Untuk dapat menginput data pasien, maka petugas pendaftaran harus memilih menu Data Pasien kemudian memilih tombol Tambah Data. Petugas pendaftaran akan diminta untuk memasukkan data-data pasien yang diperlukan. Jika data telah berhasil dimasukkan ke dalam *database*, maka aplikasi akan menampilkan pesan berhasil dan petugas pendaftaran dapat menutup menu Data Pasien. Sedangkan jika proses memasukkan data gagal, aplikasi akan menampilkan pesan gagal dan akan menampilkan kembali halaman memasukkan data.



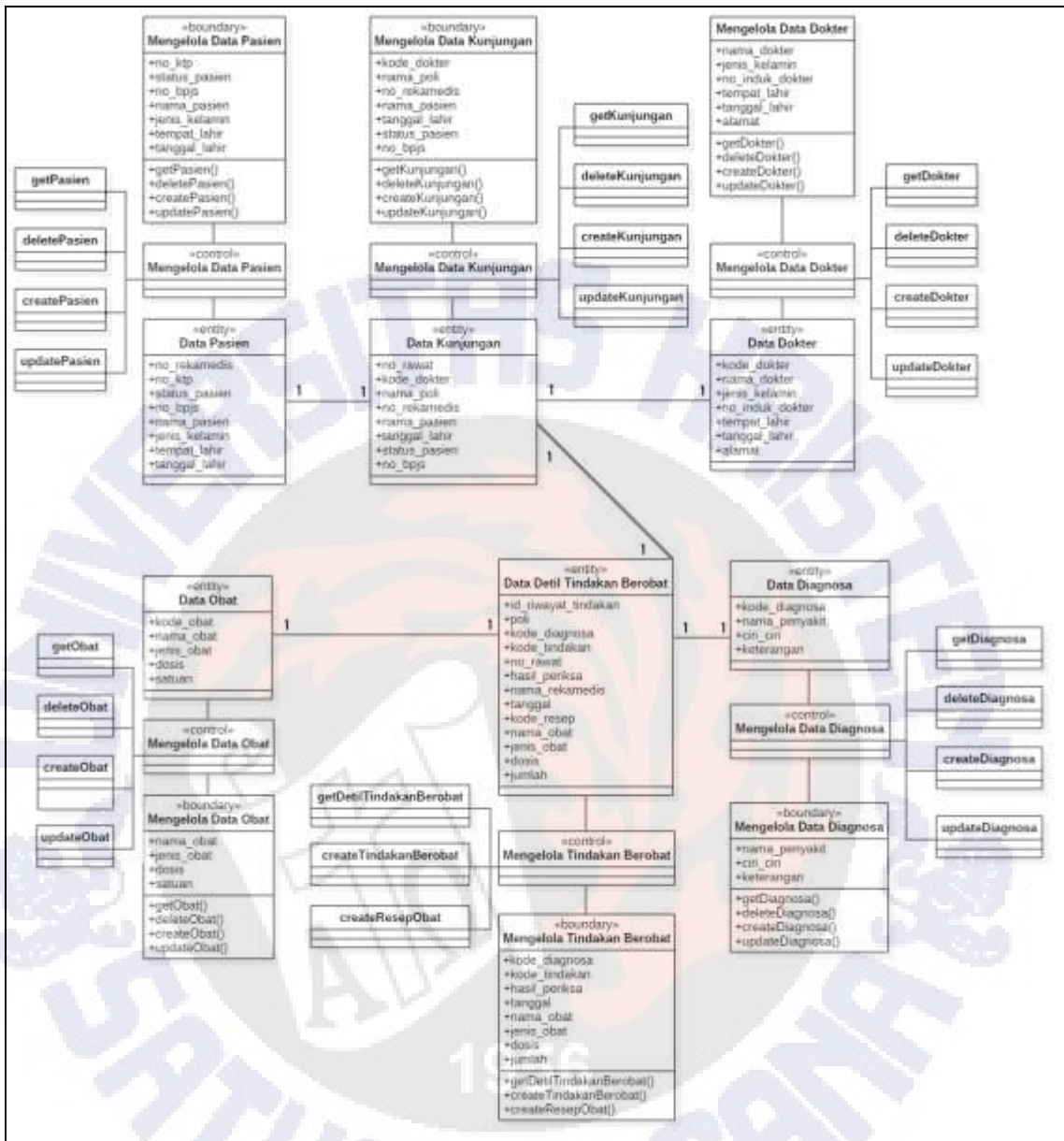
Gambar 5 Activity Diagram Menu Input Data Kunjungan

Gambar 5 menjelaskan tentang *activity diagram* pada menu Data Kunjungan. Aktivitas petugas pendaftaran untuk menginput data kunjungan pasien diawali dengan proses *login* ke dalam aplikasi. Setelah *email* dan *password* sudah terverifikasi dan valid, maka akan ditampilkan halaman menu utama pada aplikasi. Untuk dapat menginput data kunjungan, maka petugas pendaftaran harus memilih menu Data Kunjungan kemudian memilih tombol Tambah Data. Petugas pendaftaran akan diminta untuk memasukkan data-data kunjungan yang diperlukan. Jika data telah berhasil dimasukkan ke dalam *database*, maka aplikasi akan menampilkan pesan berhasil dan petugas pendaftaran dapat menutup menu Data Kunjungan. Sedangkan jika proses memasukkan data gagal, aplikasi akan menampilkan pesan gagal dan akan menampilkan kembali halaman memasukkan data.



Gambar 6 Activity Diagram Menu Input Detil Data Tindakan Berobat

Gambar 6 menjelaskan tentang *activity diagram* pada menu Data Tindakan Berobat. Daftar Data Tindakan Berobat akan muncul dengan sendirinya ketika pasien sudah dimasukkan dalam Data Kunjungan. Petugas Kesehatan tinggal memasukkan detil data tindakan berobat. Aktivitas petugas kesehatan untuk menginput detil data tindakan berobat diawali dengan proses *login* ke dalam aplikasi. Setelah *email* dan *password* sudah terverifikasi dan valid, maka akan ditampilkan halaman menu utama pada aplikasi. Untuk dapat menginput detil data tindakan berobat, maka petugas kesehatan harus memilih menu Data Tindakan Berobat kemudian memilih pasien yang sedang diperiksa. Petugas kesehatan akan diminta untuk memasukkan detil data tindakan berobat yang diperlukan. Jika data telah berhasil dimasukkan ke dalam *database*, maka aplikasi akan menampilkan pesan berhasil dan petugas kesehatan dapat menutup menu Data Kunjungan. Sedangkan jika proses memasukkan data gagal, aplikasi akan menampilkan pesan gagal dan akan menampilkan kembali halaman memasukkan data.

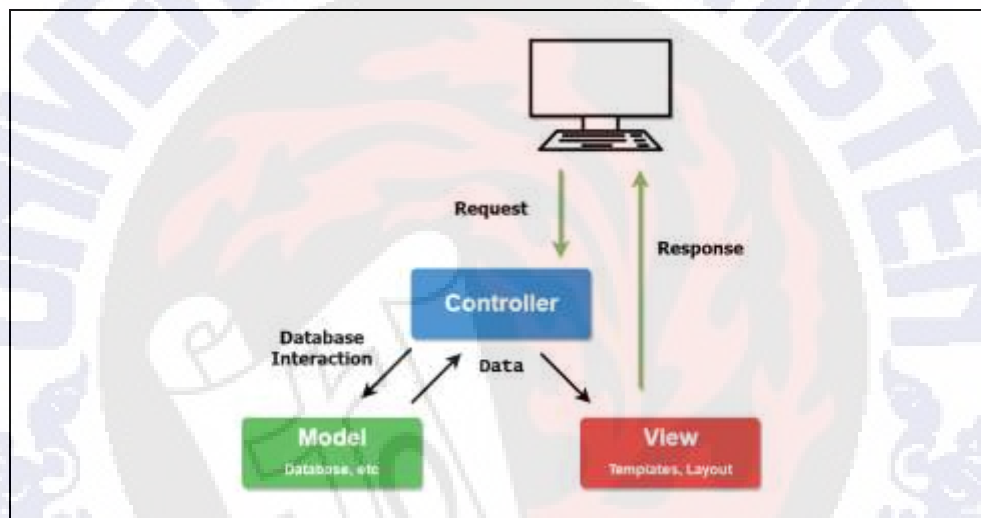


Gambar 7 Class Diagram

Gambar 7 merupakan *class diagram* aplikasi ini. Diagram ini terdiri dari enam kelas yaitu data pasien, data kunjungan, data dokter, data obat, data detil tindakan berobat, dan data diagnosa. Tiap kelas memiliki *attribut* dan *operation* masing-masing. Kelas data pasien, data kunjungan, data dokter, data obat, dan data diagnosa memiliki empat *operation* yang terdiri dari *get* data, *delete* data, *create* data, dan *update* data. Sedangkan pada kelas data detil tindakan berobat hanya memiliki tiga *operation* yaitu *get* data, *create* tindakan berobat dan *create* resep. Antar kelas juga memiliki relasi, yaitu kelas data pasien dengan data kunjungan, data kunjungan dengan data dokter, data obat dengan data detil tindakan berobat, data detil tindakan berobat dengan data diagnosa dan data kunjungan dengan data detil tindakan berobat. Kelima relasi yang ada pada kelas tersebut merupakan relasi 1 to 1.

4. Hasil dan Pembahasan

Perancangan aplikasi kemudian diimplementasikan ke dalam bentuk pengkodean (*coding*) untuk membangun sistem pelayanan Puskesmas Kaliwungu berbasis *web* dengan menggunakan *framework* CodeIgniter dan arsitektur *Model View Controller*. MVC adalah sebuah *design pattern* (pola desain) arsitektur pengembangan aplikasi yang memisahkan dan mengelompokkan beberapa kode sesuai dengan fungsinya. *Model* mewakili struktur data. Biasanya kelas *model* akan berisi fungsi yang membantu mengambil, menyisipkan, dan memperbarui informasi dalam basis data. *View* adalah informasi yang disajikan kepada pengguna. Tampilan biasanya akan menjadi halaman *web*, tetapi dalam CodeIgniter, tampilan juga bisa berupa *fragmen* halaman seperti *header* atau *footer*. Ini juga bisa berupa halaman RSS, atau "halaman" jenis lainnya. *Controller* berfungsi sebagai perantara antara *Model*, *View*, dan sumber daya lainnya yang diperlukan untuk memproses permintaan HTTP dan menghasilkan halaman *web* [13].

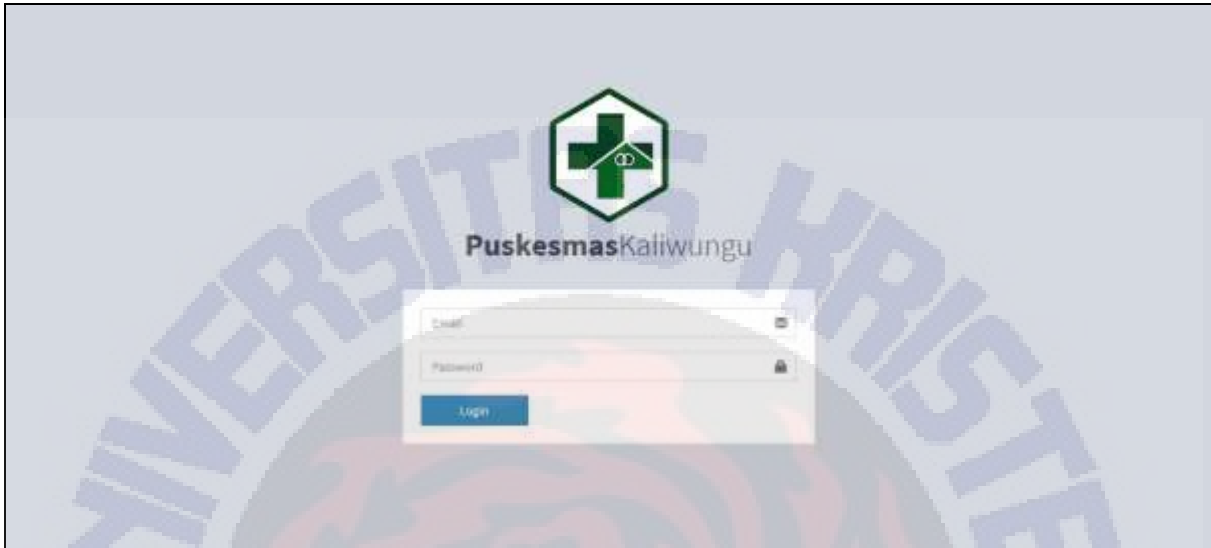


Gambar 8 Bagan MVC

Pada gambar 8 terdapat *user* mengirimkan sebuah *request* ke dalam aplikasi, *Controller* akan bertanggung jawab untuk mengambil data dari *Model* dan me-rendernya ke dalam *View* dengan menggunakan *library*, *plugin*, dan *helper* yang ada. *View* memberikan *response* dengan mengirimkan hasil *render* (*view*) ke *user* berupa sebuah halaman *web*.

Penerapan *design pattern Model View Controller* pada sistem pelayanan puskesmas kaliwungu pada bagian *Model* didefinisikan untuk digunakan untuk segala sesuatu yang berkaitan dengan *database*. *Model* berisi *function* untuk melakukan *query* mengenai informasi apa saja yang akan ditampilkan, serta untuk membuat *function insert*, *update*, serta *delete*. Bagian *view* berfungsi sebagai *presentation logic*, atau menampilkan hasil dari *model*. Tetapi *view* tidak memiliki akses langsung terhadap bagian *model*. Sehingga dibutuhkan *Controller* sebagai bagian yang mengatur hubungan antara *model* dan *view*. Pada *controller* terdapat *class* dan *function* yang memproses permintaan dari *view* ke dalam struktur data di dalam *model*. *Controller* tidak boleh berisi kode untuk mengakses basis data sebab tugas mengakses data sudah diserahkan kepada *model*. Tugas *controller* adalah menyediakan berbagai *variable* yang akan ditampilkan di *view*, memanggil *model* untuk melakukan akses ke basis data, menyediakan penanganan kesalahan (*error handling*), mengerjakan proses logika dari aplikasi serta melakukan validasi atau cek terhadap inputan. Kelebihan menerapkan *framework*

codeigniter dengan *design patern* MVC pada sitem pelayanan puskesmas kaliwungu adalah halaman *web* dapat diakses dengan lebih cepat. Sebab *application logic* terpisah dari *presentation logic* (antarmuka aplikasi), sehingga halaman *web* mengandung skrip yang sangat minimum.



Gambar 8 Halaman *Login*

Kode program 1 *Login Controller*

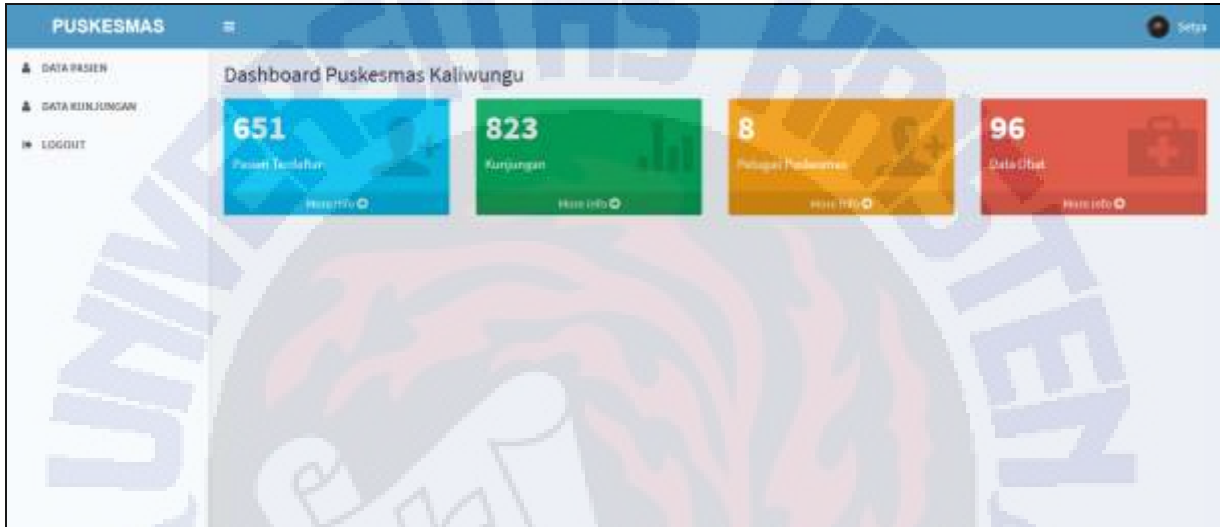
```

1. function cheklogin(){
2.
3. $email      = $this->input->post('email');
4. $password   = $this->input->post('password',TRUE);
5. $this->form_validation->set_rules('email', 'Email', 'trim|required');
6. $this->form_validation->set_rules('password', 'Password','trim|required');
7. $this->form_validation->set_message('required', '{field} wajib diisi');
8. $this->form_validation->set_error_delimiters('<div class="alert alert-danger">',
9. '</div>');
10. if ($this->form_validation->run() == FALSE) {
11. $this->index();
12. } else {
13. $hashPass = password_hash($password,PASSWORD_DEFAULT);
14. $test     = password_verify($password, $hashPass);
15. $this->db->where('email',$email);
16. $users    = $this->db->get('tbl_user');
17. if($users->num_rows()>0){
18. $user = $users->row_array();
19. if(password_verify($password,$user['password']))){
20. $this->session->set_userdata($user);
21. redirect('welcome');
22. }else{
23. $this->session->set_flashdata('message', 'Email atau password salah');
24. redirect('auth');}
25. }else{
26. $this->session->set_flashdata('message', 'Email atau Password Tidak Ditemukan');
27. redirect('auth');}}
28. }

```

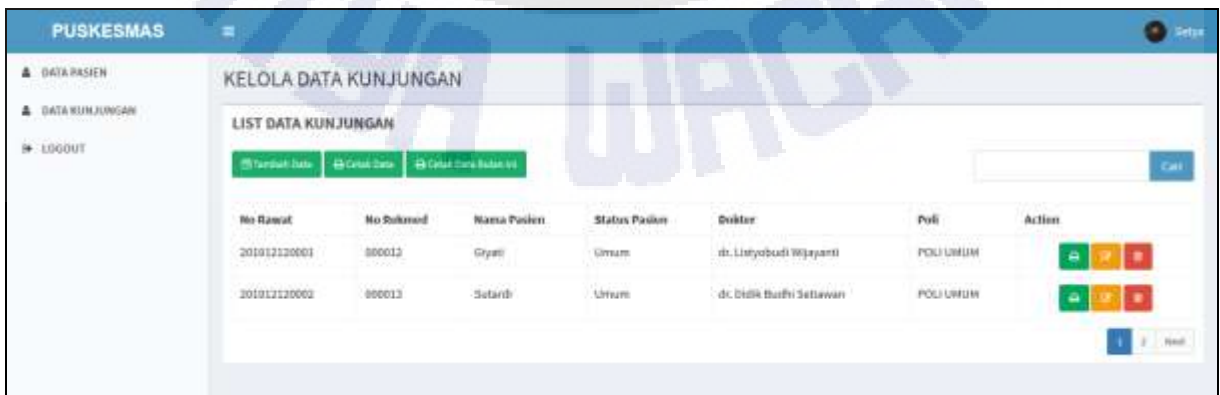
Kode program 1 adalah kode untuk menangani proses *login*. Pada baris 1 menjelaskan bahwa pendeklarasian *function* bernama *cheklogin*, dimana *function* ini akan dipanggil pada

halaman *view* pada saat *login*. Baris 2-3 mengatur inputan pada saat *login* yaitu berupa *email* dan *password*. Baris 4-7 mengatur validasi inputan pada kolom *email* dan *password*, dimana untuk melakukan *login* ke aplikasi, kolom *email* dan *password* wajib diisi dan harus dengan format *email* yang benar. Baris 8-26 menjelaskan apabila inputan sudah diisi dan sesuai dengan format yang benar, maka akan mengecek apakah kombinasi *email* dan *password* yang sudah di *hashing* terdapat di dalam *database* *tbl_user* atau tidak. Jika ditemukan maka akan dibawa ke halaman *welcome* berupa tampilan *dashboard* aplikasi, namun apabila data tidak ditemukan akan kembali ke *auth* yang merupakan halaman *login* dan muncul pesan *error*.



Gambar 9 Tampilan Home Dashboard Petugas Pendaftaran

Gambar 9 merupakan tampilan halaman utama dari sistem pelayanan Puskesmas Kaliwungu. Pada tampilan *dashboard* menampilkan *widget* berupa jumlah total pasien yang terdaftar sebagai anggota puskesmas, jumlah total kunjungan pasien, jumlah total petugas puskesmas, dan jumlah total data obat yang sudah diinputkan kedalam *database*. Disebelah kiri merupakan menu-menu yang dimiliki oleh hak akses sebagai Petugas Pendaftaran.



Gambar 10 Tampilan Kelola Data Kunjungan Pasien

Pada Gambar 10 merupakan *list* data kunjungan pasien yang berobat pada hari itu. Untuk menambah seorang pasien kedalam data kunjungan, terlebih dahulu data pasien harus diinputkan di data pasien untuk didaftar sebagai anggota pasien puskesmas. Terdapat tombol untuk melakukan tambah data, edit data, cetak data dan hapus data. Dibalik tampilan tersebut terdapat kode program yang berguna untuk menampilkan informasi tersebut ketika pengguna mengakses halaman data kunjungan pasien.

Kode program 2 Data Kunjungan Model

```

1.  $class Tbl_pendaftaran_model extends CI_Model {
2.
3.  public $table = 'tbl_pendaftaran';
4.  public $id    = 'no_rawat';
5.  public $order = 'DESC';
6.  public $no_rekamedis = 'no_rekamedis';
7.  function get_all() {
8.  $this->db->order_by($this->id, $this->order);
9.  $this->db->join('tbl_pasien', 'tbl_pendaftaran.no_rekamedis =
tbl_pasien.no_rekamedis');
10. $this->db->join('tbl_dokter', 'tbl_pendaftaran.kode_dokter_penanggung_jawab =
tbl_dokter.kode_dokter');
11. $this->db->join('tbl_poli', 'tbl_pendaftaran.id_poli = tbl_poli.id_poli');
12. return $this->db->get($this->table)->result();
13.
14. }

```

Kode Program 2 merupakan model untuk menampilkan data kunjungan. Dimana sebuah *class* bernama *Tbl_pendaftaran_model* berisi *function* bernama *get_all* yang berfungsi untuk mengambil data berupa nomer rekamedis dari tabel *tbl_pasien*, kode dokter dari tabel *tbl_dokter*, id poli dari tabel *tbl_poli*. Dimana data ditampilkan berdasarkan urutan *no_rawat* secara *descending*.

Kode program 3 Data Kunjungan View

```

1.  <table>
2.  <tr><th>No Rawat</th>
3.    <th>No Rekmed</th>
4.    <th>Nama Pasien</th>
5.    <th>Status Pasien</th>
6.    <th>Dokter</th>
7.    <th>Poli</th>
8.    <th>Action</th>
9.  </tr><?php
10. foreach ($pendaftaran_data as $pendaftaran)
11. { ??
12. <tr><td><?php echo $pendaftaran->no_rawat ?></td>
13.   <td><?php echo $pendaftaran->no_rekamedis ?></td>
14.   <td><?php echo $pendaftaran->nama_pasien ?></td>
15.   <td><?php echo $pendaftaran->status_pasien ?></td>
16.   <td><?php echo $pendaftaran->nama_dokter ?></td>
17.   <td><?php echo $pendaftaran->nama_poli ?></td>
18.   <td style="text-align:center" width="200px">
19. <?php echo anchor(site_url('pendaftaran/cetakantrian/'.$pendaftaran->no_rawat); ??
20. <?php echo anchor(site_url('pendaftaran/update/'.$pendaftaran->no_rawat); ??
21. <?php echo anchor(site_url('pendaftaran/delete/'.$pendaftaran->no_rawat); ??
22. </td>
23. </tr>
24. <?php } ??
25. </table>

```

Kode Program 3 merupakan *view* untuk menampilkan data kunjungan. Setelah di dalam *model* data kunjungan sudah diatur data apa saja yang akan diambil dari *database*, dibagian *view* ditampilkan kedalam sebuah tabel. Dipanggil semua *field-field* yang akan ditampilkan.

Kode program 4 Data Kunjungan Controller

```

1. class Pendaftaran extends CI_Controller {
2.     public function index(){
3.         $q = urldecode($this->input->get('q', TRUE));
4.         $start = intval($this->input->get('start'));
5.         if ($q <> '') {
6.             $config['base_url'] = base_url() . 'index.php/pendaftaran/index?q=' . urlencode($q);
7.             $config['first_url'] = base_url() . 'index.php/pendaftaran/index?q=' . urlencode($q);
8.         } else { $config['base_url'] = base_url() . 'index.php/pendaftaran/index';
9.             $config['first_url'] = base_url() . 'index.php/pendaftaran/index';
10.        $config['per_page'] = 10;
11.        $config['page_query_string'] = TRUE;
12.        $config['total_rows'] = $this->Tbl_pendaftaran_model->total_rows($q);
13.        $pendaftaran = $this->Tbl_pendaftaran_model->get_all($config['per_page'], $start, $q);
14.        $config['full_tag_open'] = '<ul class="pagination pagination-sm no-margin pull-right">';
15.        $config['full_tag_close'] = '</ul>';
16.        $this->load->library('pagination');
17.        $this->pagination->initialize($config);
18.        $data = array( 'pendaftaran_data' => $pendaftaran,
19.            'q' => $q,
20.            'pagination' => $this->pagination->create_links(),
21.            'total_rows' => $config['total_rows'],
22.            'start' => $start,);
23.        $this->template->load('template','pendaftaran/tbl_pendaftaran_list', $data);}

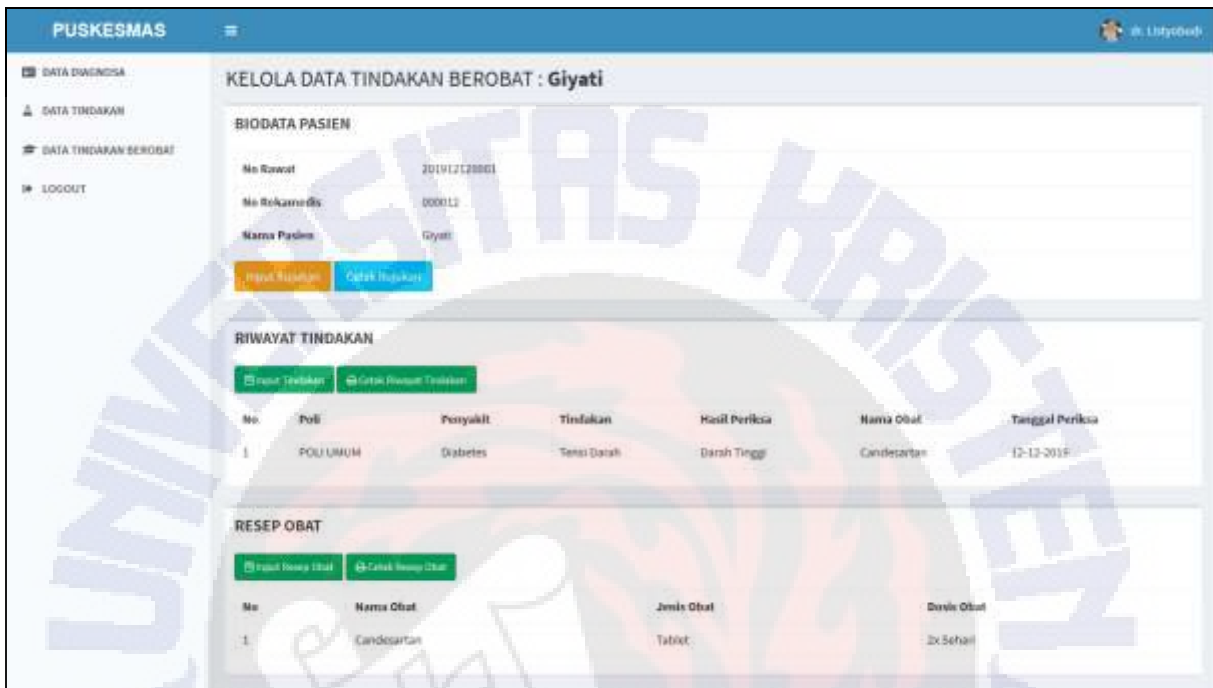
```

Kode Program 4 merupakan *controller* untuk menampilkan data kunjungan. Setelah di dalam *model* data kunjungan sudah diatur data apa saja yang akan diambil dari *database*, di *view* ditampilkan kedalam sebuah tabel. Di *Controller* diatur supaya *model* dan *view* dapat terhubung. Diatur beberapa *configurasi* dengan fungsi *base_url()* adalah fungsi bantuan dari *url helper*, fungsi *base_url()* digunakan untuk membangun *url* untuk mengakses *resource* yang berada di direktori *root web*. Diatur mengenai jumlah data yang akan ditampilkan dalam satu halaman, yakni hanya 10 baris data. Diatur juga tentang fitur *pagination*, diberikan link untuk bisa mengakses ke halaman lainya. Proses menghubungkan *model* dan *view* terdapat pada baris 15, didalam *controller* memanggil fungsi *getall* yang berisi *query* ke *database* di dalam *model*. Pada baris kode ke-26, menunjukkan fungsi *index* yang ada di dalam *controller* akan ditampilkan di halaman *tbl_pendaftaran_list*, yang merupakan *view* data kunjungan.

No Reg	No Rawat	No Rekrud	Nama Pasien	Dokter Penanggung Jawab	Poli	Action
0001	201912120001	000012	Oiyati	dr. Lityobodi Wijayanti	POLI UMUM	[Green] [Red]
0002	201912120002	000013	Subardi	dr. Dikri Budhi Setiawan	POLI UMUM	[Green] [Red]

Gambar 11 Tampilan Kelola Data Tindakan Berobat

Gambar 11 menampilkan tampilan *list* data tindakan berobat yang dilakukan oleh Petugas Kesehatan. *List* data tindakan berobat ini sudah otomatis muncul sendiri, ketika pasien sudah dimasukan ke data kunjungan saat akan berobat. Data yang ditampilkan adalah data pasien, data dokter yang akan menangani, serta poli yang dituju pasien.



Gambar 12 Tampilan Detil Data Tindakan Berobat

Gambar 12 setelah pasien sudah didaftarkan didalam data kunjungan, maka secara otomatis data muncul di *list* data kunjungan berobat. Selanjutnya petugas kesehatan dapat memasukan hasil pemeriksaan di halaman detil data tindakan berobat ini. Di halaman detil data tindakan berobat menampilkan secara singkat biodata pasien, riwayat rindakan yang pernah didapatkan dan resep obat yang akan didapatkan saat pemeriksaan dengan petugas kesehatan. Petugas kesehatan dapat menginputkan tindakan yang dilakukan selama pemeriksaan, kemudian data tersebut akan masuk kedalam *list* riwayat tindakan. Obat yang diberikan kepada pasien setelah diinputkan akan muncul di bagian resep obat. Namun apabila hasil pemeriksaan mengharuskan pasien untuk di rujuk ke rumah sakit, petugas kesehatan juga dapat menginputkan data rujukan dan kemudian dapat dicetak.

Pengujian sistem ini dilakukan dengan menggunakan teknik pengujian *Black Box Testing*. *Black box testing* berfokus pada persyaratan fungsional perangkat lunak. Pengujian *black box* adalah bagian integral dari pengujian kebenaran tetapi idenya tidak terbatas pada pengujian kebenaran saja. *Black box testing* melengkapi teknik *white box testing* dan cenderung mengungkap kelas kesalahan yang berbeda dengan metode *white box* [14]. Untuk rincian pengujian dapat dilihat pada tabel berikut:

- a. Pengujian untuk bagian yang dapat diakses oleh *user* admin pada sistem pelayanan puskesmas kaliwungu dapat dilihat pada Tabel 1.

Tabel 1 Pengujian *Black Box* Admin

Aktivitas dan Event	Input	Output yang diharapkan	Output	Status Pengujian
Admin <i>login</i>	<i>email</i> dan <i>password</i>	a. Jika berhasil akan masuk ke menu utama b. Jika gagal akan menampilkan notifikasi bahwa <i>email</i> atau <i>password</i> tidak sesuai.	Admin berhasil <i>login</i> sistem dengan <i>email</i> dan <i>password</i> yang benar	Valid
Admin tambah data Pengguna	Nama lengkap, Email, <i>password</i> , <i>Level user</i> , Status, Foto profil	a. Jika data pengguna baru berhasil atau valid dimasukkan maka akan dilakukan proses penambahan data pada daftar pengguna b. Jika data pengguna baru tidak valid maka muncul pesan <i>error</i>	Admin berhasil menambahkan data pengguna dengan data yang valid	Valid
Admin tambah data Pegawai	NIP, Nama pegawai, Jenis kelamin, NPWP, Tempat lahir, Tanggal lahir, Jabatan	a. Jika data pegawai baru berhasil atau valid dimasukkan maka akan dilakukan proses penambahan data pada daftar pegawai b. Jika data pegawai baru tidak valid maka muncul pesan <i>error</i>	Admin berhasil menambahkan data Pegawai dengan data yang valid	Valid
Admin tambah data Dokter	SIP, Nama dokter, Jenis kelamin, Tempat lahir, Tanggal lahir, Alamat, Bagian poli	a. Jika data Dokter baru berhasil atau valid dimasukkan maka akan dilakukan proses penambahan data pada daftar dokter b. Jika data dokter baru tidak valid maka muncul pesan <i>error</i>	Admin berhasil menambahkan data Dokter dengan data yang valid	Valid
Admin tambah data POLI	Nama Poli, Ruang Poli	a. Jika data poli baru berhasil atau valid dimasukkan maka akan dilakukan proses penambahan data pada daftar poli b. Jika data poli baru tidak valid maka muncul pesan <i>error</i>	Admin berhasil menambahkan data POLI dengan data yang valid	Valid
Admin tambah data <i>Level User</i>	Nama <i>Level User</i>	a. Jika data <i>level user</i> baru berhasil atau valid dimasukkan maka akan dilakukan proses penambahan data pada daftar <i>level user</i> b. Jika data <i>level user</i> baru tidak valid maka muncul pesan <i>error</i>	Admin berhasil menambahkan data <i>level user</i> dengan data yang valid	Valid
Admin tambah data Hak Akses	Memberi Tanda Centang	Jika data hak akses baru dcentang maka akan dilakukan proses penambahan data pada daftar hak akses	Admin berhasil tambah data hak akses	Valid

- b. Pengujian untuk bagian yang dapat diakses *user* Petugas Pendaftaran mata pelajaran pada sistem pelayanan puskesmas kaliwungu dapat dilihat pada Tabel 2.

Tabel 2 Pengujian *Black Box* Petugas Pendaftaran

Aktivitas dan Event	Input	Output yang diharapkan	Output	Status Pengujian
Petugas pendaftaran login	<i>Email</i> dan <i>Password</i>	a. Jika berhasil akan masuk ke menu utama b. Jika gagal akan menampilkan notifikasi bahwa <i>email</i> atau <i>password</i> tidak sesuai.	Petugas pendaftaran berhasil <i>login</i> dengan <i>email</i> dan <i>password</i> yang valid	Valid
Petugas pendaftaran tambah data Pasien	No KTP, Status pasien, No BPJS (jika peserta), Nama, Jenis kelamin, Tempat lahir, Tanggal lahir, Alamat	a. Jika data pasien baru berhasil atau valid dimasukkan maka akan dilakukan proses penambahan data pada daftar pasien b. Jika data pasien baru tidak valid maka muncul pesan <i>error</i>	Petugas pendaftaran berhasil tambah data pasien dengan data yang valid	Valid
Petugas pendaftaran tambah data Kunjungan Pasien	Dokter, Poli tujuan, No rekamedis, Nama pasien, Tanggal lahir, Status pasien, No BPJS (jika peserta)	a. Jika data kunjungan pasien baru berhasil atau valid dimasukkan maka akan dilakukan proses penambahan data pada daftar data kunjungan b. Jika data kunjungan pasien baru tidak valid maka muncul pesan <i>error</i>	Petugas pendaftaran berhasil tambah data kunjungan pasien dengan data yang valid	Valid

- c. Pengujian untuk bagian yang dapat diakses *user* Petugas Kesehatan pada sistem pelayanan puskesmas kaliwungu dapat dilihat pada Tabel 3.

Tabel 3 Pengujian *Black Box* Petugas Kesehatan

Aktivitas dan Event	Input	Output yang diharapkan	Output	Status Pengujian
Petugas kesehatan login	<i>Email</i> dan <i>Password</i>	a. Jika berhasil akan masuk ke menu utama b. Jika gagal akan menampilkan notifikasi bahwa <i>email</i> atau <i>password</i> tidak sesuai.	Petugas kesehatan berhasil <i>login</i> dengan <i>email</i> dan <i>password</i> yang valid	Valid
Petugas kesehatan tambah data Diagnosa	Kode diagnosa, Nama penyakit, Ciri-ciri, Keterangan, Ciri-ciri umum	a. Jika data diagnosa baru berhasil atau valid dimasukkan maka akan dilakukan proses penambahan data pada daftar diagnosa b. Jika data diagnosa baru tidak valid maka akan	Petugas kesehatan berhasil menambahkan data diagnosa dengan data yang valid	Valid

muncul pesan <i>error</i>				
Petugas kesehatan tambah data Tindakan	Kode, Nama Tindakan, Ditangani oleh	a. Jika data tindakan baru berhasil atau valid dimasukkan maka akan dilakukan proses penambahan data pada daftar diagnosa b. Jika data tindakan baru tidak valid maka akan muncul pesan <i>error</i>	Petugas kesehatan berhasil menambahkan data tindakan dengan data yang valid	Valid
Petugas kesehatan tambah data Tindakan berobat	Nama penyakit, Nama tindakan, Hasil periksa, Nama obat	a. Jika data tindakan berobat baru berhasil atau valid dimasukkan maka akan dilakukan proses penambahan data pada daftar tindakan berobat b. Jika data tindakan berobat baru tidak valid maka akan muncul pesan <i>error</i>	Petugas kesehatan berhasil menambahkan data tindakan berobat dengan data yang valid	Valid
Petugas kesehatan tambah data Resep Obat	Nama obat, Jenis obat, Aturan Pakai, Jumlah obat	a. Jika data resep obat baru berhasil atau valid dimasukkan maka akan dilakukan proses penambahan data pada daftar resep obat b. Jika data resep obat baru tidak valid maka akan muncul pesan <i>error</i>	Petugas kesehatan berhasil menambahkan data resep obat dengan data yang valid	Valid
Petugas kesehatan tambah data Rujukan	Nama penyakit, Diagnosa, Nama RS, Poli tujuan	a. Jika data resep obat baru berhasil atau valid dimasukkan maka akan dilakukan proses penambahan data pada daftar resep obat b. Jika data resep obat baru tidak valid maka akan muncul pesan <i>error</i>	Petugas kesehatan berhasil menambahkan data resep obat dengan data yang valid	Valid

d. Pengujian untuk bagian yang dapat diakses *user* Petugas Apotek pada sistem pelayanan puskesmas kaliwungu dapat dilihat pada Tabel 4.

Tabel 4 Pengujian *Black Box* Petugas Apotek

Aktivitas dan Event	Input	Output yang diharapkan	Output	Status Pengujian
Petugas apotek <i>login</i>	<i>Email</i> dan <i>Password</i>	a. Jika berhasil akan masuk ke menu utama b. Jika gagal akan menampilkan notifikasi bahwa <i>email</i> atau <i>password</i> tidak sesuai.	Petugas apotek berhasil <i>login</i> dengan <i>email</i> dan <i>password</i> yang valid	Valid
Petugas apotek tambah data Obat	Kode Obat, Nama Obat, Jenis Obat, Aturan pakai Obat, Jenis Satuan	a. Jika data obat baru berhasil atau valid dimasukkan maka akan dilakukan proses penambahan data pada daftar obat b. Jika data obat baru tidak valid maka akan muncul pesan <i>error</i>	Petugas apotek berhasil menambahkan data obat dengan data yang valid	Valid

Berdasarkan hasil pengujian yang dilakukan diperoleh hasil bahwa masing-masing fungsi pada aplikasi sistem pelayanan puskesmas kaliwungu sudah sesuai dengan yang diharapkan. Pengujian *beta* merupakan pengujian yang dilakukan oleh suatu kelompok yang tidak terlibat dalam pembuatan sistem, pengujian ini bertujuan untuk mengetahui apakah sistem telah memenuhi kebutuhan pengguna. Untuk mengetahui hasil dari pengujian *beta* digunakan kuesioner yang diberikan kepada pengguna sistem. Tanggapan dari masing-masing responden yang telah mengisi kuesioner kemudian dihitung dengan menggunakan *skala likert*. *Skala likert* menggunakan beberapa butir pertanyaan untuk mengukur perilaku individu dengan merespon 5 titik pilihan pada setiap butir pertanyaan, sangat setuju, setuju, tidak memutuskan, tidak setuju, dan sangat tidak setuju [15]. Pertanyaan dalam kuesioner dapat dilihat pada Tabel 5.

Tabel 5 Tabel Daftar Pertanyaan Kuesioner

Nomor	Pertanyaan
1	<i>Website</i> sistem pelayanan puskesmas kaliwungu mudah digunakan?
2	Pada saat pengoperasian sistem aplikasi apakah mudah dipahami?
3	<i>Website</i> sistem pelayanan puskesmas kaliwungu berjalan sesuai dengan kebutuhan, yakni menghasilkan output berupa pelayanan yang lebih efisien?
4	<i>Website</i> sistem pelayanan puskesmas kaliwungu membantu proses pengolahan data puskesmas?
5	Penggunaan sistem berbasis <i>web</i> ini mempermudah proses pelayanan puskesmas kaliwungu?

Jawaban kuesioner dihitung menggunakan persamaan skala *likert* dengan tujuan untuk mengetahui persentase dari masing-masing jawaban kuesioner. Berdasarkan hasil pengujian yang telah dilakukan sehingga dapat disimpulkan bahwa pertanyaan pertama 82% responden memberi pendapat aplikasi ini mudah digunakan. Sedangkan berdasarkan pertanyaan ke dua 80% responden berpendapat aplikasi mudah dipahami. Berdasarkan pertanyaan ketiga 74% responden mengatakan aplikasi ini berjalan sesuai dengan kebutuhan. Sebanyak 78% responden berpendapat bahwa aplikasi membantu proses pengolahan data pelayanan puskesmas. Berdasarkan pertanyaan terakhir 82% responden berpendapat bahwa aplikasi mempermudah proses pelayanan puskesmas.

5. Kesimpulan

Berdasarkan pengujian aplikasi yang telah dilakukan, maka dapat diambil kesimpulan bahwa semua fungsi dari aplikasi berjalan sesuai dengan rancangan yang telah dibuat. Dalam membangun sistem pelayanan puskesmas kaliwungu menggunakan *framework* CodeIgniter yang membantu mempercepat dalam proses pengembangan. Sistem pelayanan puskesmas kaliwungu berhasil mengoptimalkan proses pelayanan kesehatan pada puskesmas kaliwungu menjadi lebih efisien. Serta mampu membantu petugas puskesmas kaliwungu dalam mengolah data menjadi lebih mudah.

Sistem masih dapat dikembangkan dengan dapat melakukan *bridging* dengan Pcare dari BPJS. Sehingga dalam proses pelayanan pada puskesmas kaliwungu dapat dilakukan dengan lebih efisien lagi, sebab proses pelayanan pasien baik peserta BPJS dan *non* peserta BPJS dapat ditangani dalam satu sistem yang sama.

6. Daftar Pustaka

- [1] No, P. M. K. (75). tahun 2014 tentang Pusat Kesehatan Masyarakat. *Kementerian Kesehatan RI, Jakarta*.
- [2] Apa Itu CodeIgniter dan Keunggulannya, <https://www.logique.co.id/blog/2018/08/13/codeigniter-dan-keunggulannya>, diakses 21 November 2019
- [3] Permatasari, D. (2014). *Perancangan Sistem Informasi Layanan Kesehatan Puskesmas Ngemplak Kabupaten Boyolali* (Doctoral dissertation, Universitas Muhammadiyah Surakarta).
- [4] Yuniar, E., & Muslim, M. H. (2018). SISTEM INFORMASI LAYANAN KESEHATAN DENGAN MENGGUNAKAN CODEIGNITER PADA PUSKESMAS BULULAWANG. *Antivirus: Jurnal Ilmiah Teknik Informatika*, 12(1).
- [5] Syukron, A., & Hasan, N. (2015). Perancangan Sistem Informasi Rawat Jalan Berbasis Web Pada Puskesmas Winong. *Bianglala Informatika*, 3(1).
- [6] Sutabri, Tata. 2012. Konsep Dasar Informasi. Yogyakarta: Andi
- [7] Mengenal Bahasa Pemrograman PHP, <https://idcloudhost.com/mengenal-bahasa-pemrograman-php>, diakses 2 Juni 2019
- [8] Mengenal Apa itu Framework CodeIgniter, <https://idcloudhost.com/panduan/mengenal-apa-itu-framework-codeigniter>, diakses 2 Juni 2019
- [9] Razzoli, F. (2014). *Mastering MariaDB*. Packt Publishing Ltd.
- [10] Bartholomew, D. (2014). *MariaDB cookbook*. Packt Publishing Ltd.
- [11] Witham, M., Bender, I., & Gomes, R. Comparative Analysis of MariaDB's Performance Efficiency as a Suitable Replacement for MySQL.
- [12] Sugiyono. (2008). *Metode penelitian pendidikan:(pendekatan kuantitatif, kualitatif dan R & D)*. Alfabeta.
- [13] Model-View-Controller, <https://codeigniter.com/userguide3/overview/mvc.html>, diakses 12 Desember 2019
- [14] Khan, M. E. (2011). Different approaches to black box testing technique for finding errors. *International Journal of Software Engineering & Applications*, 2(4), 31.
- [15] Likert, R. (1932). A technique for the measurement of attitudes. *Archives of psychology*.