BAB 2
TINJAUAN PUSTAKA

2.1. Landasan Teori

Landasan teori akan menjelaskan beberapa teori yang berkaitan dengan permasalahan yang di bahas sebagai pemahaman sistem yang akan dibangun pada pembangunan sistem pencegahan penumpukan pasien. Beberapa teori terkait dengan pembangunan perangkat lunak ini adalah konsep yang digunakan, perangkat lunak, perangkat keras dan bahasa pemograman yang dibutuhkan dalam Perancangan Sistem Pencegahan Penumpukan Pasien Dengan *People Counter* dan kunci pintu pintar menggunakan mikrokontroler.

2.1.1. Sistem

Menurut Jogiyanto, Sistem berasal dari bahsa Yunani (*Sustema*) adalah suatu kesatuan yang yang terdari komponen,atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi dan energi. Sistem juga merupakan kesatuan bagian-bagian yang saling terhubung yang berada dalam suatu wilayah. Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran yang tertentu [3].

2.1.2. Sistem Antrian

Sistem antrian adalah hasil pengembangan teori antrian, yang mengatur pelayanan berurutan sesuai kedatangan, untuk mencapai kinerja efektif dan efisien, sebagai solusi masalah antrian di berbagai tempat yang menggunakan prosedur antrian. Sistem antrian akan mempunyai ciri khas tersendiri darai setiap sistem yang ada di perusahaan, sesuai dengan ruang tunggu pelayanan yang ada, dan tidak akan pernah ada di tempat lain. Sistem antrian atau nama lainya adalah mesin antrian, memang sebaiknya dibuat secara khusus untuk memenuhi semua kebutuhan dari sebuah perusahaan atau instansi, karena sistem harus bisa mengetahui cara melayani orang-orang yang memerlukan jasa anda, dengan cara yang terbaik. Sistem antrian mempunyai pengaruh besar terhadap instansi usaha atau bisnis yang anda kelola, karena sistem antrian adalah garda depan perusahaan yang selalu dituntut tampil sempurna, karena keterkaitan dengan Marketing, serta Customer Relationship Management [4].

2.1.3. Pasien

Pasal 1 Undang-undang No. 29 Tahun 2004 Tentang Praktik Kedokteran menjelaskan definisi pasien adalah setiap orang yang melakukan konsultasi masalah kesehatannya untuk memperoleh pelayanan kesehatan yang diperlukan baik secara langsung maupun tidak langsung kepada dokter [5].

Dalam sistem yang akan dibangun pasien akan menjadi objek yang nanti akan menjadi acuan sensor akan menghitung jumlah orang yang masuk. Hal awal yang dilakukan pasien adalah memasuki ruang tunggu melewati pintu masuk, saat melewati proses itu sensor membaca masuknya pasien sebagai jumlah banyak pasien yang berada di ruang tunggu.

2.1.4. Kunci Pintu

Kunci pintu berfungsi sebagai alat untuk menghentikan gerakan pintu, yang mana pintu tersebut tidak akan bisa terbuka ketika kunci pintu dipergunakan. Sebagai alat rumah tangga dan juga alat musik, kunci adalah jenis alat pengancing yang berfungsi untuk mencegah terbukanya daun pintu atau penutup lainnya dari kedudukan semula [6].

Kunci pintu yang digunakan dalam sistem ini adalah pendorongan yang dilakukan oleh sebuah alat yang nanti akan di tempatkan di pintu. Pintu yang digunakan di Klinik Prima Jaya ini adalah pintu *swing,* jenis pintu yang biasa digunakan di setiap rumah. Pintu *swing* akan ditempelkan dengan suatu alat yang dapat mendorong pintu maju dan mundur. Mikrokontroler arduino digunakan sebagai pengontrol dari alat itu tersebut

2.1.5. Mikrokontroler

Mikrokontroler merupakan komponen utama yang berfungsi sebagai pengatur pergerakan motor dan pengolah data yang dihasilkan oleh komparator sebagai bentuk keluaran dari sensor. Mikrokontroler merupakan sebuah *processor* yang digunakan untuk kepentingan kontrol. Meskipun mempunyai bentuk yang jauh lebih kecil dari suatu komputer pribadi dan komputer *mainframe*, mikrokontroler dibangun dari elemen – elemen dasar yang sama. Seperti komputer, mikrokontroler adalah alat yang mengerjakan beberapaa instruksi yang diberikan kepadanya. Artinya, bagian terpenting dan utama dari suatu sistem terkomputerisasi adalah program itu sendiri. Program ini menginstruksikan komputer untuk melakukan tugas yang lebih kompleks yang diinginkan.

2.1.5.1. Sejarah Mikrokontroler

Sejarah mikrokontroler tidak terlepas dari sejarah mikroprosesor dan lomputer. Diawali dengan ditemukanya mikroprosesor, kemudiann ditemukan komputer, setelah itu ditemukan mikrokontroler.

Berikut ini sejarah mikrokontroler:

1. Tahun 1971, Intel 4004 adalah Mikroprosesor pertama intel 4004 dibuat dan dikembangkan oleh intel (*Integrated Electronics*). Intel membuat mikroprosesor 4004 menggunakan 2550 transistor. Intel 4004 merupakan mikroprosesor 4 bit. Kemudian pada tahun 1974, intel membuat mikroprosesor generasi kedua (Intel 8008), intel 8008 merupakan mikroprosesor 8 bit.
2. Tahun 1972 TMS1000 adalah mikrokontroler pertama. TMS1000 merupakan mikrokontroler 4-bit. Mikrokontroler TMS1000 dibuat oleh Gary Boone dari *Texas Instrument*.
3. Tahun 1974, beberapa pabrikan IC menawarkan mikroprosesor dan pengendali menggunakan mikroprosesor. Mikroprosesor yang ditawarkan pada saat itu yaitu Intel 8080, 8085, Motorola 6800, Signetics 6502, Zilog Z80, dan Texas Instrument 9900 (16 bit).
4. Tahun 1976, dibuat Intel 8048, yang merupakan mikrokontroler Intel pertama.
5. Tahun 1978, mikrokontroler 16 bit menjadi umum digunakan yaitu Intel 8086,i Motorola 68000 dan Zlog Z8000. Mikroprosesor yang dikembangkan termasuk 32 bit *device* seperti Intel Pentium.
6. Tahun 1980, Intel 8051 atau lebih dikenal dengan mikrokontroler MCS51 adalah CISC 8 bit.
7. Tahun 1996, mikrokontroler Atmel AVR dibuat oleh Atme. Atmel AVR adalah salah satu mikrokontroler yang banyak digunakan sampai saat ini. AVR adalah mikrokontroler RISC (*Reduce Instrument Set Computing*) 8 bit.

2.1.5.2. Cara Kerja Mikrokontroler

Prinsip kerja mikrokontroler adalah sebagai berikut:

1. Berdasaarkan nilai yang berada pada register Program Counter, mikrokontroler mengambil data dari ROM dengan *address* sebagaimana nilai yang ada pada Program *Counter*. Selanjutnya Program *Counter* ditambah nilainya dengan 1 (*increment*) secara otomatis. Data yang diambil tersebut adalah urutan intruksi program pengendali mikrokontroler yang sebelumnya telah dibuat oleh pemakai.

2. Proses pengerjaan bergantung pada jenis intruksi: bisa membaca, mengubaha nilai – nilai pada *register*, RAM, isi *port*, atau melakukan pembacaan dan dilanjutkan dengan pengubahan data.
3. Program *Counter* telah berubah nilainya (baik karena penambahan otomatis sebagimana pada langkah 1 diatas atau karena pengubahan pada langkah 2). Selanjutnya yang dilakukan mikrokontroler adalah mengulang kembali siklus ini pada langkah 1 dan seterusnya.

Dari pengertian diatas dapat disimpulkan bahwa pada dasarnya untuk kerja mikrokontroler sangatlah bergantung pada urutan intruksi yang dijalankanya, yaitu program yang ditulis di ROM [7].

2.1.6. Arduino

Arduino adalah perangkat elektronik yang berfungsi seperti mikrokontroler dan bersifat *open source*. Arduino memiliki prosesor keluarga Atmel AVR. Arduino memiliki perangkat lunak dengan bahasa pemograman yang spesifik. Arduino juga memiliki *sofware* kompilasi sendiri yang bersifat *open source* dan dapat diunduh di *website arduino.cc*. Sehingga pengguna dapat mengembangkan sendiri *board* arduino sesuai dengan keinginan.

Board arduino mempunyai kemampuan untuk membaca masukan data digital dan data analog. *Board* Arduino juga mempunyai kemampuan untuk mengeluarkan data digital dari analog. Bahasa pemrogaman yang digunakan mempunyai *sintaks* bahasa C. Pengguna Arduino dapat mengembangkan beberapa proyek sederhana (motor, *led* data *serial*) dan proyek yang kompleks (*Internet of Things*, printer 3D dan aplikasi canggih lainya). Arduino dapat berjalan diatas komputer dengan sistem oprasi mac, windows dan linux. Perkembangan arduino cukup pesat karena didukung oleh adanya komunitas *online* arduino.

2.1.6.1. Sejarah Arduino

Arduino dikembangkan dari thesis Hernando Barragan pada tahun 2004, seorang mahasiswa asal Kolombia. Judul thesisnya yaitu "Arduino-Revolusi *Open Hardware*". Arduino diawali dari ruangan kelas *Interactive Design Institute* di Ivrea (IDII), pada tahun 2005 di Ivera, italia. Arduino ditemukan oleh Massimo Banzi dan David Cuartielles dengan tujuan awal yaitu untuk membantu para siswa membuat perangkat desain dan interaksi dengan harga yang murah dibandingkan dengan perangkat lain yang tersedia pada saat itu, seperti *BASIC Stamp* yang harganya cukup mahal bagi pelajar pada saat itu.

Arduino berasal dari bahasa Italia yang berarti teman yang berani. Pada bulan Mei 2011,arduino sudah terjual lebih dari 3000 unit arduino saat ini sudah menjadi salah satu *platform* OSHW (*Open Source Hardware*). Arduino memiliki banyak jenis, berikut ini adalah jenis – jenis arduino yaitu:

1. Arduino Uno (R3)
2. Arduino Due
3. Arduino Mega2560
4. Arduino Leonardo
5. Arduino Fio
6. Arduino LilyPad
7. Arduino Nano
8. Arduino Mini
9. Arduino Micro
10. Arduino Ethernet
11. Arduino Espora
12. Arduino Robot

Kelebihan – kelebihan dari *board* arduino diantaranya adalah :

1. Tidak perlu perangkat *chip programmer* karena didalamnya memiliki *bootloader* yang akan menangani program yang di-*upload* dari *computer*.
2. Bahasa pemogrammanya relatif mudah (bahasa C), dan *sofware* arduino mudah dioprasikan karena berbentuk GUI (*Graphical User Interface*), IDE (*Integreated Development Envirotment*), memiliki *liblary* yang cukup lengkap serta geratis dan *Open Source*.
3. Komunikasi *serial* dan komunikasi untuk *upload* program menggunakan jalur yang sama yaitu melalui jalur USB (atau komunikasi *serial*), sehingga membutuhkan banyak kabel [8].

2.1.7. Arduino Uno

Arduino Uno adalah *board* mikrokontroler berbasis ATmega328 (datasheet). Memiliki 14 *pin input* dari output digital dimana 6 *pin input* tersebut dapat digunakan sebagai output PWM dan 6 pin input analog, 16 MHz osilator kristal, koneksi USB, jack *power*, ICSP *header*, dan tombol reset. Untuk mendukung mikrokontroler agar dapat digunakan, cukup hanya menghubungkan Board Arduino Uno ke komputer dengan menggunakan kabel USB atau listrik dengan AC yang-ke adaptor-DC atau baterai untuk menjalankannya. Nama "Uno" berarti satu dalam bahasa Italia, untuk menandai

peluncuran Arduino 1.0. Uno dan versi 1.0 akan menjadi versi referensi dari Arduino. Uno adalah yang terbaru dalam serangkaian board USB Arduino, dan sebagai model referensi untuk *platform* Arduino, untuk perbandingan dengan versi sebelumnya, lihat indeks board Arduino.Uno Arduino dapat diaktifkan melalui koneksi USB atau dengan catu daya eksternal (otomatis). Eksternal (non-USB) daya dapat berasal baik dari AC-ke adaptor-DC atau baterai. Adaptor ini dapat dihubungkan dengan menancapkan plug jack pusat-positif ukuran 2.1mm konektor POWER. [9].

Dalam hal ini arduino uno akan menjadi mikrokontroler yang nanti akan menjadi pengontrol segala fungsi yang ada pada sistem. Semua komponen utama akan di hubungkan dengan arduino uno, sehingga arduino uno dapat menerima dan mengirimkan data.



***Sumber Gambar : [online] Arduino Uno* [2]**

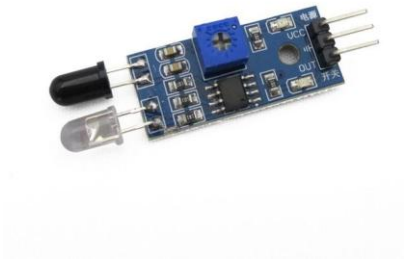
Gambar 2. 1 Arduino Uno

2.1.8. IR *module*

Infra red (IR) detektor atau sensor infra merah adalah komponen elektronika yang dapat mengidentifikasi cahaya infra merah (infra red, IR). Sensor infra merah atau detektor infra merah saat ini ada yang dibuat khusus dalam satu modul dan dinamakan sebagai IR *Detector* Photomodules. IR *Detector* Photomodules merupakan sebuah chip detektor inframerah digital yang di dalamnya terdapat fotodiode dan penguat (amplifier). Bentuk dan Konfigurasi Pin IR *Detector* Photomodules TSOP. Konfigurasi pin infra *red* (IR) *receiver* atau penerima infra merah tipe TSOP adalah *output* (*Out*), Vs (VCC +5 volt DC), dan *Ground* (GND). Sensor penerima inframerah TSOP (TEMIC Semiconductors Optoelectronics Photomodules) memiliki fitur-fitur utama yaitu fotodiode dan penguat dalam satu chip, keluaran aktif rendah, konsumsi daya rendah, dan mendukung logika TTL dan CMOS. Detektor infra merah atau sensor inframerah jenis TSOP (TEMIC Semiconductors Optoelectronics Photomodules) adalah penerima inframerah yang telah

dilengkapi filter frekuensi 30-56 kHz, sehingga penerima langsung mengubah frekuensi tersebut menjadi logika 0 dan 1. Jika detektor inframerah (TSOP) menerima frekuensi carrier tersebut, maka pin keluarannya akan berlogika 0. Sebaliknya, jika tidak menerima frekuensi carrier tersebut, maka keluaran detektor inframerah (TSOP) akan berlogika 1. Sistem sensor infra merah pada dasarnya menggunakan infra merah sebagai media untuk komunikasi data antara *receiver* dan *transmitter*. Sistem akan bekerja jika sinar infra merah yang dipancarkan terhalang oleh suatu benda yang mengakibatkan sinar infra merah tersebut tidak dapat terdeteksi oleh penerima. Keuntungan atau manfaat dari sistem ini dalam penerapannya antara lain sebagai pengendali jarak jauh, alarm keamanan, otomatisasi pada sistem[10].

Dalam hal ini IR *module* akan menjadi sensor utama yang akan mendeteksi objek. Objek yang akan dideteksi adalah pasien. Selain pasien, objek yang akan terdeteksi adalah berbagai macam benda yang menghalangi cahaya *infra red.* Dalam kasus ini perancangan IR *module* akan dibuat seminimal mungkin mengurangi kesalahan dalam pendetesian objek.



***Sumber Gambar :[online] Sensor Pendeteksi objek* [*2*]**

Gambar 2. 2 IR Module

2.1.9. LED

Lampu LED atau kepanjangannya *Light Emitting Diode* adalah suatu lampu indikator dalam perangkat elektronika yang biasanya memiliki fungsi untuk menunjukkan status dari perangkat elektronika tersebut. Lampu LED terbuat dari plastik dan dioda semikonduktor yang dapat menyala apabila dialiri tegangan listrik rendah (sekitar 1.5 volt DC). Bermacam-macam warna dan bentuk dari lampu LED, disesuaikan dengan kebutuhan dan fungsinya. Berikut adalah contoh lampu LED gambar 2.3 [11].

Dalam sistem yang akan dibangun lampu LED akan diguanakan sebagai peringatan dini kepada dokter perihal jumlah pasien yang ada di ruang tunggu dan pasien

darurat yang mengunjungi klinik. Sistem ini akan menggunakan 3 lampu LED yaitu, lampu LED berwarna merah sebagai peringatan kepada dokter bahwa ada pasien darurat datang, lampu LED kuning sebagai peringatan dini kepada dokter bahwa pasien yang berada di ruang tunggu hampir mencapai maksimal, dan lampu LED hijau sebagai peringatan dini bahwa pasien yang berad di ruang tunggu sudah mencapai maksimal.



***Sumber Gambar : [online] LED 5mm* [2]**

Gambar 2. 3 LED

2.1.10. LCD

Display elektronik adalah salah satu komponen elektronika yang berfungsi sebagai tampilan suatu data, baik karakter, huruf ataupun grafik. LCD (Liquid Cristal Display) adalah salah satu jenis display elektronik yang dibuat dengan teknologi CMOS logic yang bekerja dengan tidak menghasilkan cahaya tetapi memantulkan cahaya yang ada di sekelilingnya terhadap front-lit atau mentransmisikan cahaya dari back-lit. LCD (*Liquid Cristal Display*) berfungsi sebagai penampil data baik dalam bentuk karakter, huruf, angka ataupun grafik[12].

Dalam sistem yang akan di bangung LCD yang akan digunakan adalah LCD 16x2 yang nanti akan di pasangkan dalam purwarupa dalam perancangan sisitem *people counter*. LCD akan menampilkan beberapa informasi untuk pasien seperti, jumlah pasien yang berada di ruang tunggu dan jumlah pasien yang dapat diperiksa oleh dokter.



***Sumber Gambar : [online] LCD 16x2* [2]**

Gambar 2. 4 LCD

2.1.11. Ethernet *Shield*

Ethernet Shield adalah modul yang berfungsi menghubungkan Arduino board dengan jaringan internet,karna itu berdasar pada Wiznet W5100 ethernet chip untuk menghubungkan dan menggunakan modul hingga dapat terkoneksi internet cukup mudah, hanya membutuhkan waktu beberapa menit. Dalam ethernet sendiri terdapat slot mikro SD yang berbungsi sebagai tempat penyimpanan file sedangkan untuk mengakses mikro SD card mengunakan library SD ,untuk jenis arduino board yang bisa di pasangkan dengan ethernet shield W5100 yaitu arduino uno dan mega [13].

Dalam sistem ini ethernet *shield* akan digunakan sebagai penghubung sistem ke dalam jaringan internet. Sensor akan mengirimkan data pasien yang berkunjung perharinyake dalam *website* yang telah dibangun dalam sistem. Dokter yang mengatur batasan pasien juga dapat mengatur batasan pasien melalui jaringan internet.



***Sumber Gambar : [online] Ethernet Shield*[2]**

Gambar 2. 5 Ethernet *Shield*

2.1.12. Motor *Driver*

Motor *Driver* adalah sebuah motor DC yang dilengkapi rangkaian kendali dengan sistem closed feedback yang terintegrasi dalam motor tersebut. Pada motor *driver* posisi putaran sumbu dari motor akan diinformasikan kembali ke rangkaian kontrol yang ada di dalam motor *servo*. Motor *driver* disusun dari sebuah motor DC, gearbox, variabel resistor (VR) atau potensiometer dan rangkaian kontrol. Potensiometer berfungsi untuk menentukan batas maksimum putaran sumbu (axis) motor servo. Sedangkan sudut dari sumbu motor *driver* diatur berdasarkan lebar pulsa yang pada pin kontrol motor servo [14].

Dalam sistem ini motor *driver* akan di gunakan sebagai kunci pintu. Motor *driver* akan mendorong atau menarik pintu sesuai dengan jumlah pasien yang berada di ruang tunggu. Motor *driver* yang ditempatkan hanya di pintu depan saja. Karena proses

pengantrian harus bermula dari masuknya pasien melalui pintu depan. Servo yang digunakan dalam purwa rupa adala motor driver 5v.



***Sumber Gambar : [online] Motor Driver* [2]**

Gambar 2. 6 Motor *Driver*

2.1.13. *Printer Thermal* TTL

Printer thermal adalah merupakan salah satu jenis printer dari keluarga printer impact, dimana printer impact ini biasanya sering digunakan sebagai printer untuk kasir pembelanjaan dan mesin fax. Proses pencetakan dari printer thermal ini dengan menggunakan gulungan kertas. Dimana proses bekerjanya dengan mengambil gulungan kertas tersebut kemudian berubah menjadi gelap saat dipanaskan. Saat mencetak kertas akan ditarik bersebelahan dengan kepala cetak yang berisi pemanas eletronik[15].

Dalam sistem yang akan dibagun printer akan dijadikan sarana percetakan karcis antrian. Karcis antrian akan tercetak dan keluar ketika pasien sudah menekan tombol tiket. Karcis antrian akan terus mencetak sampai mencapai batas kuota pasien yang ditentukan. Setelah pasien melewati batas, maka printer akan mencetak peringantan kepada pasien bahwa karcis antrian sudah habis, sehingga pasien diminta untuk mendatangi klinik atau rumah sakit lain.

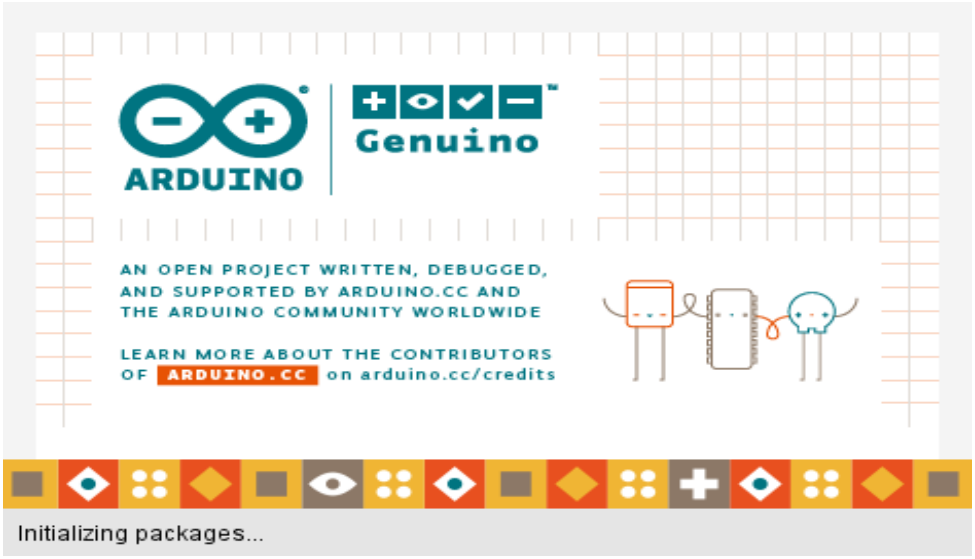


***Sumber Gambar : [online] printer TTL for Arduino* [2]**

Gambar 2. 7 Printer Thermal TTL

2.1.14. Arduino IDE

Arduino IDE adalah pengendali mikro *single-board* yang bersifat *open-source* diturunkan dari *platform Wiring*, dirancang untuk memudahkan penggunaan eletronik dalam berbagai bidang, *hardware*-nya menggunakan prosesor Atmel AVR dan *software*-nya memiliki bahasa pemrograman C++ yang sederhana dan fungsi-fungsinya yang lengkap, sehingga arduino mudah dipelajari oleh pemula. Berikut adalah tampilan awal Arduino IDE gambar 2.8 [16].



Gambar 2. 8 Tampilan Awal Arduino IDE

2.1.15. Pemrograman Berorientasi Objek

Metodologi berorientasi objek adalah suatu strategi pembangunan peralngkan lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek yang berisi data dan operasi yang diberlakukan terhadapnya. Metodologi berorientasi objek merupakan suatu cara bagaimana sistem perangkat lunak dibangun melalui pendekatan objek secara sistematis. Metode berorientas objek didasarkan pada penerapan prinsip-prinsip pengelolaan kompleksitas. Metode berorientasi objek meliputi rangkaian aktivitas analisis berorientasi objek, perancangan berorientasi objek, pemrograman berorientasi objek, dan pengujian berorientasi objek

Keuntungan menggunakan metodologi berorientasi objek adalah sebagai berikut :

1. Meningkatkan produktivitas

   Karena kelas dan objek yang ditemukan dalam suatu masalah masih dapat dipakai ulang untuk masalah lainnya yang melibatkan objek tersebut (*reusable*).

2. Kecepatan pengembangan

   Karena sistem yang dibangun dengan baik dan benar pada saat analisis dan perancangan akan menyebabkan berkurangnya kesalahan pada saat pengodean.

3. Kemudahan pemeliharaan

Karena dengan model objek, pola-pola yang cenderung tetap dan stabil dapat dipisahkan dan pola-pola yang mungkin sering berubah-ubah.

4. Adanya konsistensi

Karena sifat pewarisan dan penggunaan notasi yang sama pada saat analisis, perancangan maupun pengkodean.

5. Meningkatkan kualitas perangkat lunak

Karena pendekatan pengembangan lebih dekat dengan dunai nyata dan adanya konsistensi pada saat pengembangannya, perangkat lunak yang dihasilkan akan mampu memenuhi kebutuhan pemakai serta mempunyai sedikit kesalahan [17].

2.1.16. Konsep Dasar Berorientasi Objek

Pendekatan berorientasi objek merupakan suatu teknik atau cara pendekatan dalam melihat permasalahan dan sistem (sistem perangkat lunak, sistem informasi, atau sistem lainnya). Pendekatan berorientasi objek akan memandang sistem yang akan dikembangkan sebagai suatu kumpulan objek yang berkorespondensi dengan objek-objek dunia nyata. Ada banyak cara untuk mengabstraksikan dan memodelkan objek-objek tersebut, mulai dan abstraksi objek, kelas, hubungan antar kelas sampai abstraksi sistem. Saat mengabstraksikan dan memodelkan objek, data dan proses-proses yang dipunyai oleh objek akan dienkapsulasi (dihubungkan) menjadi suatu kesatuan. Dalam rekayasa perangkat lunak, konsep pendekatan berorientasi objek dapat diterapkan pada tahap analisis, perancanggan, pemrograman, dan pengujian perangkat lunak. Ada berbagai teknik yang dapat digunakan pada masing-masing tahap tersebut, dengan aturan dan alat bantu pemodelan tertentu. Sistem berorientasi objek merupakan sebuah sistem yang dibangun dengan berdasarkan metode berorientasi objek adalah sebuah sistem yang komponennya dibungkus (dienkapsulasi) menjadi kelompok daata dan fungsi. Setiap komponen dalam sistem tersebut dapat mewarisi atribut dan sifat dan komponen lainnya, dan dapat berinteraksi satu sama lain. Berikut ini adalah beberapa konsep dasar yang harus dipahami tentang metodologi berorientasi objek:

1. Kelas (*Class*)

Kelas adalah sekumpulan objek-objek dengan karakteristik yang sama. Kelas merupakan definisi statis dan himpunan objek yang sama yang mungkin lahir atau diciptakan dan kelas tersebut. Sebuah kelas akan mempunyai sifat (*atribut*), kelakuan (metode/operasi), hubungan (*relationship*) dan arti. Suatu kelas dapat diturunkan dan kelas yang lain, dimana *atribut* dan kelas semula dapat diwariskan ke kelas yang baru. Secara tekniks kelas adalah sebuah struktur dalam pembuatan perangkat lunak. Kelas merupakan bentuk struktur pada kode program yang menggunakan metodologi berorientasi objek.

2. Objek (*object*)

Objek adalah abstraksi dari sesuatu yang mewakili dunia nyata benda, manusia, satua organisasi, tempat, kejadian, struktur, status, atau hal-hal lain yang bersifat abstra. Objek merupakan suatu entitas yang mampu menyimpan informasi (status) dan mempunyai operasi (kelakuan) yang dapat diterapkan atau dapat berpengaruh pada status objeknya. Objek mempunyai siklus hidup yaitu diciptakan, dimanipulasi, dan dihancurkan. Secara teknis, sebuah kelas saat program dieksekusi makan akan dibuat sebuah objek. Objek dilihat darisegi teknis adalah elemen pada saat *runtime* yang akan diciptakan, dimanipulasi, dan dihancurkan saat eksekusi sehinga sebuah objek hanya ada saat sebuah program dieksekusi. Jika masih dalam bentuk kode, disebut sebagai kelas jadi pada saat *runtime* (saat sebuah program dieksekusi), yang kita punya adalah objek, di dalam teks program yang kita lihat hanyalah kelas.

3. Metode (*method*)

Operasi atau metode pada sebuah kelas hampir sama dengan fungsi atau prosedur pada metodologi struktural. Sebuah kelas boleh memiliki lebih dari satu metode atau operasi. Metode atau operasi yang berfungsi untuk memanipulasi objek itu sendiri. Operasi atau metode merupakan fungsi atau transformasi yang dapat dilakukan terhadap objek atau dilakukan oleh objek. Metode atau operasi dapat berasal dari *event*, aktifitas atau aksi keadaan, fungsi, atau kelakuan dunnia nyata. Contoh metode atau operasi misalnya *Read*, *Write*, *Move*, *Copy*, dan sebagainya.

4. Atribut (*attribute*)

Atribut dari sebuah kelas adalah variabel global yang dimiliki sebuah kelas. Atribut dapat berupa nilai atau elemen-elemen data yang dimiliki oleh objek dalam kelas objek. Atribut dipunyai secara individual oleh sebuah objek, misalnya berat, jenis, nama, dan sebagainya.

5. Abstraksi (*abstraction*)

Prinsip untuk merepresentasikan dunia nyata yang kompleks menjadi satu bentuk model yang sederhana dengan mengabaikan aspek-aspek lain yang tidak sesuai dengan permasalahan.

6. Enkapsulasi (*encapsulation*)

Pembungkusan atribut data dan layanan (operasi-operasi) yang dipunyai objek untuk menyembunyikan implementasi dan objek sehingga objek lain tidak mengetahui cara kerja-nya.

7. Pewarisan (*inheritance*)

Mekanisme yang memungkinkan satu objek mewarisi sebagian atau seluruh definisi dan objek lain sebagai bagian dan dirinya.

8. Antarmuka (*interface*)

Antarmuka sangat mirip dengan kelas, tapi tanpa atribut kelas dan memiliki metode yang dideklarasikan tanpa isi. Deklarasi metode pada sebuah *interface* dapat diimplementasikan oleh kelas lain.

9. *Reusability*

Pemanfaatan kembali objek yang sudah didefinisikan untuk suatu permasalahan pada permasalahan lainnya yang melibatkan objek tersebut.

10. Generalisasi dan Spesialisasi

Menunjukkan hubungan antara kelas dan objek yang umum dengan kelas dan objek yang khusus. Misalnya kelas yang lebih umum (generalisasi) adalah kendaraan darat dan kelas khususnya (spesialisasi) adalah mobil, motor, dan kereta.

11. Komunikasi Antar objek

Komunikasi antar objek dilakukan lewat pesan (*message*) yang dikirim dari satu objek ke objek lainnya.

12. Polimorfisme (*polymorphism*)

Kemampuan suatu objek digunakan di banyak tujuan yang berbeda dengan nama yang sehingga menghemat baris program.

13. *Package*

Package adalah sebuah kontainer atau kemasan yang dapat digunakan untuk mengelompokkan kelas-kelas sehingga memungkinkan beberapa kelas yang bernama sama disimpan dalam *package* yang berbeda.

2.1.17. Unified Modeling Language (UML)

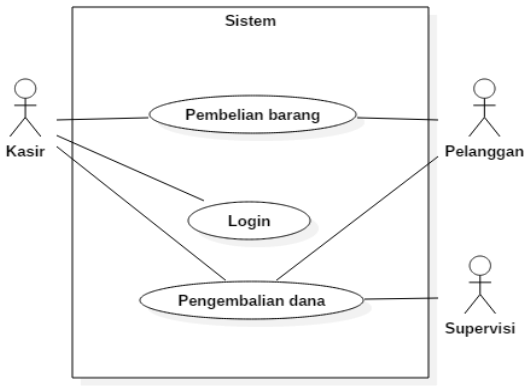
UML adalah sekumpulan alat yang digunakan untuk melakukan abstraksi terhadap sebuah sistem atau perangkat lunak berbasis objek. UML merupakan singkatan dari *Unified Modeling Language*. UML juga menjadi salah satu cara untuk mempermudah pengembangan aplikasi yang berkelanjutan. Aplikasi atau sistem yang tidak terdokumentasi biasanya dapat menghambat pengembangan karena *developer* harus melakukan penelusuran dan mempelajari kode program [18].

UML juga dapat menjadi alat bantu untuk transfer ilmu tentang sistem atau aplikasi yang akan dikembangkan dari satu *developer* ke *developer* lainya. Tidak hanya antar *developer* terhadap orang bisnis dan siapapun dapat memahami sebuah sistem dengan adanya UML. Berikut ini merupakan beberapa bagian dari UML adalah sebagai berikut :

1. *Use Case Diagram*

Use Case atau *Use Case Diagram* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use Case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *Use Case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi - fungsi itu.

Syarat penamaan pada *Use Case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *Use Case* yaitu pendefinisian apa yang disebut aktor dan *Use Case*.

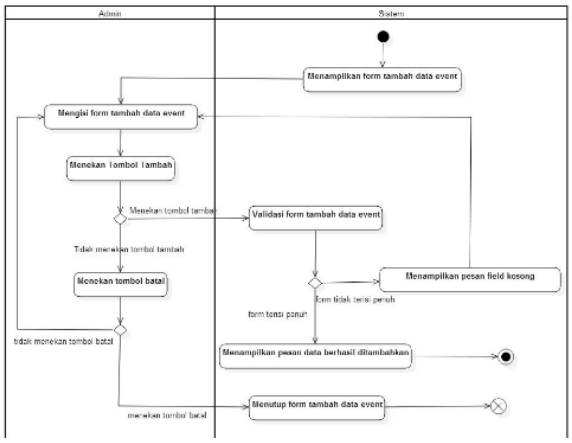


Gambar 2. 9 *Use Case Diagram*

2. *Activity Diagram*

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut :

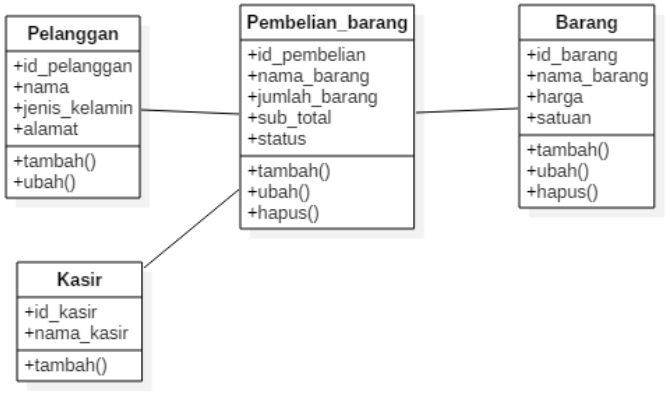
1. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
2. Urutan atau pengelompokan tampilan dari sistem/*user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.



Gambar 2. 10 *Activity Diagram*

3. *Class Diagram*

Class diagram menggambarkan struktur dan deskripsi *class*, *package* (paket) dan objek beserta hubungan satu sama lain seperti *containment* (penahanan), pewarisan, asosiasi dan lain-lain.



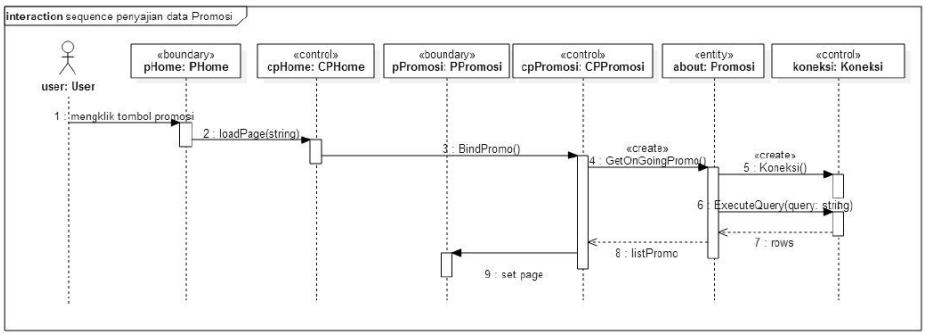
Gambar 2. 11 *Class Diagram*

4. *Sequence Diagram*

Sequence Diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram *sequence* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu.

Banyaknya diagram *sequence* yang harus digambar adalah sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram *sequence* sehingga semakin banyak *use case* yang didefinisikan maka diagram *sequence* yang harus dibuat juga semakin banyak.

Penomoran pesan berdasarkan urutan interaksi pesan. Penggambaran letak pesan harus berurutan, pesan yang lebih atas dari lainnya adalah pesan yang berjalan terlebih dahulu.



Gambar 2. 12 *Sequence Diagram*

2.1.18. Javascript

JavaScript adalah bahasa pemrograman *web* yang bersifat *Client Side Programming Language*. *Client Side Programming Language* adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh *client*. Aplikasi *client* yang dimaksud merujuk kepada *web browser* seperti Google Chrome dan Mozilla Firefox.

Bahasa pemrograman Client Side berbeda dengan bahasa pemrograman *Server Side* seperti PHP, dimana untuk *server side* seluruh kode program dijalankan di sisi *server*. Untuk menjalankan JavaScript, kita hanya membutuhkan aplikasi *text editor* dan *web browser*. JavaScript memiliki fitur: *high-level programming language, client-side, loosely tiped* dan berorientasi objek.

2.1.18.1 Fungsi JavaScript Dalam Pemograman Web

JavaScript pada awal perkembangannya berfungsi untuk membuat interaksi antara user dengan situs *web* menjadi lebih cepat tanpa harus menunggu pemrosesan di web server. Sebelum javascript, setiap interaksi dari *user* harus diproses oleh *web server*. JavaScript tidak hanya berguna untuk validasi *form*, namun untuk berbagai keperluan yang lebih modern. Berbagai animasi untuk memperindah halaman *web*, fitur *chatting*, efek-efek modern, games, semuanya bisa dibuat menggunakan JavaScript. Akan tetapi karena sifatnya yang dijalankan di sisi *client* yakni di dalam *web browser* yang digunakan oleh pengunjung situs, *user* sepenuhnya dapat mengontrol eksekusi JavaScript.[19].

2.1.18.2 JSON

JSON (*JavaScript Object Notation*) adalah format pertukaran data (*lightweight data-interchange format*), mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (*generate*) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemprograman JavaScript, Standar ECMA-262 Edisi ke-3 – Desember 1999. JSON merupakan format teks yang tidak bergantung pada bahasa pemprograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran-data [20].

Kelebihan dari JSON adalah sebagai berikut:

1. Ukuran lebih kecil dari XML, efeknya transfer data lebih hemat *resource* terutama *bandwidth*.
2. JSON adalah format data bawaan Javascript.
3. Format JSON lebih sederhana
4. Liblary JSON ada di setiap bahasa pemograman.

2.1.19. MySQL

Database merupakan *databse server* yang *open source* yang cukup populer. Dengan berbagai keunggulan yang dimiliki, membuat *sofware* ini banyak digunakanoleh para praktisi untuk membangun suatu *project*. Adanya fasilitas API (*Application Progamming interface*) yang dimiliki oleh MySQL memungkinkan berncam-macam aplikasi komputer yang ditulis dengan berbagai bahas pemograman dapat mengakses basis data MySQL.

MySQL pertama kali dirilis oleh seorang programmer *database* bernama Michael Widenius. MySQL adalah program *database* yang mampu mengirim dan menerima data dengan sangat cepat dari *multi user*. MySQL memiliki dua bentuk lisensi, yaitu *free sofware* dan *shareware*. Keunggulan MySQL memiliki keunggulan dalam menyediakan berbagai fasilitas atau fitur yang dapat digunakan oleh berbagai macam *user* [21].

2.1.20. *Web Server*

Web server merupakan *software* yang memberikan layanan data, berfungsi menerima permintaan HTTP atau HTTPS dari *client* yang dikenal dengan *browser web* dan mengirimkan kembali hasilnya dalam bentuk halaman-halaman *web* yang umumnya berbentuk dokumen HTML, konsep *web server* antara lain:

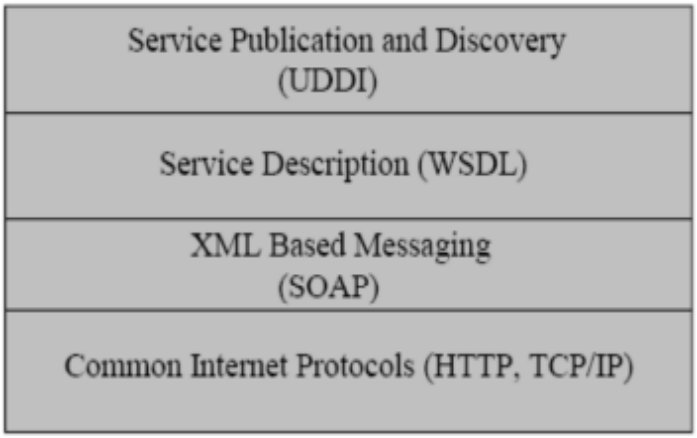
1. *Web server* merupakan mesin aplikasi atau *software* yang beroperasi dalam medistribusikan *web page* ke *user*, tentu saja sesuai dengan permintaan *user*.
2. Hubungan antara *web server* dan *browser* internet merupakan gabungan atau jaringan komputer yang berada diseluruh dunia. Setelah terhubung secara fisik, *protocol* TCP/IP (*networking protocol*) yang memungkinkan semua komputer dapat berkomunikasi antar satu dengan lainnya. Pada saat aplikasi *browser* meminta data *web page* ke *server* maka instruksi permintaan data oleh *browser* tersebut dikemas dalam TCP yang merupakan *protocol transport* dan dikirim ke alamat yang merupakan *protocol* berikutnya yaitu *hyper text transfer protocol* (HTTP). Data yang diparsing dari *browser* ke *web server* disebut sebagai HTTP *request* yang meminta halaman *web* dan kemudian *web server* akan mencari data HTML yang dibutuhkan dan dikemas dalam TCP *protocol* kemudian dikirim kembali ke *browser*. Data yang dikirim dari *server* ke *browser* disebut sebagai HTTP response. Jika data yang diminta oleh *browser* tidak ditemukan pada *web server* maka akan menampilkan *error* pada *web page* yaitu *Error: 404 Page Not Found* [22].

2.1.21. *Web Service*

Web service merupakan suatu komponen *software* yang merupakan *selfcontaining*, aplikasi modular *self-describing* yang dapat dipublikasikan, dialokasikan, dan dilaksanakan pada *web*. *Web service* adalah teknologi yang mengubah kemampuan internet dengan menambahkan

kemampuan *transactional web*, yaitu kemampuan *web* untuk saling berkomunikasi dengan pola *program-to-program* (P2P). Fokus *web* selama ini didominasi oleh komunikasi *program-to-user* dengan interaksi *business-to-consumer* (B2C), sedangkan *transactional web* akan didominasi oleh *program-to-program* dengan interaksi *business-to-business*. gambar 2.16 Merupakan blok bangunan *web service* yang mana menyediakan fasilitas komunikasi jarak jauh antara dua aplikasi yang merupakan *layer* arsitektur *web service* [23].

1. Layer 1 : protokol internet standar yang digunakan sebagai sarana transportasi adalah HTTP dan TCP/IP.
2. Layer 2 : *Simple Object Access Protocol* (SOAP) berbasiskan XML dan digunakan untuk pertukaran informasi antar sekelompok layanan.
3. Layer 3 : *Web service Definition Language* (WSDL) digunakan untuk mendiskripsikan attribute layanan.
4. Layer 4 : *Universal Description, Discovery and Integration*, yang mana merupakan direktori pusat untuk deskripsi layanan.



Gambar 2. 13 Blok Bangun *Web Service*

2.1.22. Sampling Insidental

Penarikan sampel dilakukan dengan cara sampling insidental. Sampling insidental adalah sampling teknik penentuan sampel berdasarkan kebetulan, yaitu siapa saja yang secara kebetulan atau insidental bertemu dengan peneliti dapat digunakan sabagai sampel, bila dipandang orang yang kebetulan ditemui itu cocok sebagai sumber data.

2.1.23. Skala Likert

Skala Likert di gunakan untuk mengukur sikap, pendapat, dan persepsi seseorang atau sekelompok orang tentang fenomena sosial. Dalam penelitian, fenomena sosial ini telah ditetapkan secara spesifik oleh peneliti, yang selanjutnya disebut sebagai variabel

penelitian.Dengan skala likert, maka variabel yang akan diukur dijabarkan menjadi indikator variabel. Kemudian indikator tersebut dijadikan sebagai titik tolak untuk menyusun item-item instrumen yang dapat berupa pernyataan atau pertanyaan. Jawaban setiap item instrumen yang menggunakan skala likert mempunyai gradasi dari sangat positif sampai negatif , yang dapat berupa kata-kata antara lain [24]:

a. Sangat Setuju
b. Setuju
c. Ragu-Ragu
d. Tidak Setuju
e. Sangat Tidak Setuju

Untuk keperluan analisis kuantitatif, maka jawaban itu dapat di beri skor, misalnya :

1. Setuju/selalu/sangat positif di beri skor 5
2. Setuju/sering/positif diberi skor 4
3. Ragu-ragu/kadang-kadang/netral diberi skor 3
4. Tidak setuju/hampir tidak pernah/negati diberi skor 2
5. Sangat tidak setuju/tidak pernah diberi skor 1

Berikut ini adalah rumus yang dipakai dalam pengambilan data sampel dari data populasi yang telah dikumpulkan. Rumus yang dipakai adalah rumus slovin, rumus ini dapat menetukan data sampel yang dibutuhkan dalam populasi.

$$n = \frac{N}{1 + N\,\alpha^2}$$

Keterangan:
n = jumlah sampel
N = jumlah populasi
α = margin error

2.1.24. *Business Process Modelling Notation* (BPMN)

BPMN adalah standar untuk memodelkan proses bisnis dan proses-proses secara efisien. BPMN menyediakan notasi yang dapat dengan mudah dipahami oleh semua pengguna bisnis, termasuk juga analis bisnis yang menciptakan draf awal dari proses sampai pengembang teknis yang bertanggung jawab untuk mengimplementasikan teknologi yang digunakan untuk menjalankan proses-proses tersebut [25