# Teknologi Informasi

**USULAN PENELITIAN**



# PERANCANGAN SISTEM PAKAR UNTUK DIAGNOSA PENYAKIT GONORE DENGAN MENGGUNAKAN METODE NAÏVE BAYES BERBASIS WEB

TIM PENGUSUL :

Ketua : Teri Ade Putra, S.Kom, M.Kom / NIDN : 1001059001

Anggota :

Raja Ayu Mahessya, S.Kom, M.Kom/NIDN:1023119001

Pradani Ayu Widya Purnama, S.Kom, M.Kom/ NIDN:   
 /1008039301

Reza Suriadinata

**UNIVERSITAS PUTRA INDONESIA “YPTK” PADANG 2021**

**Lembaran Pengesahan Proposal**

**LEMBARAN PENGESAHAN PENGAJUAN PROPOSAL PENELITIAN PENGABDIAN MASYARAKAT**

**UNIVERSITAS PUTRA INDONESIA “YPTK” PADANG**

Judul

# PERANCANGAN SISTEM PAKAR UNTUK DIAGNOSA PENYAKIT GONORE DENGAN MENGGUNAKAN METODE NAÏVE BAYES BERBASIS WEB

Ketua Pelaksana

Nama : Teri Ade Putra S.Kom,M.Kom

NIDN : 1001059001

Pangkat/golongan : Penata(3c)

Jabatan Pendidikan : Dosen

Fakultas : Ilmu Komputer

Jurusan : Teknik Informatika

Email :Teriadeputra@upiyptk.ac.id

No. Telp/HP

Personalia

Anggota Pelaksana : 2org

Anggota I : Raja Ayu Mahessya, S.Kom, M.kom

Anggota II : Pradani Ayu Widya Purnama, S.Kom, M.Kom

Jangka waktu : Satu tahun ( 1 )

Bentuk Kegiatan : Pelatihan dan Uji Kompetensi

Sifat Kegiatan : Pendidikan dan Latihan

Sumber dana : Mandiri

Mengetahui,

Padang, 12 Juli 2021

Mengetahui,

Ketua Jurusan Ketua Pelaksana

,

Eka Praja Wiyata Mandala,S.Kom,M.kom Teri Ade Putra,S.kom,M.Kom

NIDN : 1014088502 NIDN : 1001059001

Mengetahui,

Dekan Fakultas Ilmu Komputer

Universitas Putra Indonesia “YPTK” Padang

Dr.Yuhandri,S.Kom,M.Kom

NIDN : 1015057301

DAFTAR ISI

HALAMAN JUDUL i

HALAMAN PENGESAHAN ii

[DAFTAR ISI iii](#_TOC_250070)

[RINGKASAN iv](#_TOC_250069)

[BAB I PENDAHULUAN 1](#_TOC_250066)

* 1. Latar Belakang 1
  2. Tujuan Penelitian 3
  3. [Manfaat Penelitian 3](#_TOC_250065)
  4. [Batasan Masalah 3](#_TOC_250064)

[BAB II LANDASAN TEORI 6](#_TOC_250060)

* 1. [Rekayasa Perangkat Lunak 6](#_TOC_250059)
     1. [Defenisi Perangkat Lunak 6](#_TOC_250058)
     2. [Defenisi Rekayasa Perangkat Lunak 7](#_TOC_250057)
     3. [Proses Perangkat Lunak 7](#_TOC_250056)
     4. [Kriteria Perangkat Lunak 8](#_TOC_250055)
     5. [Tantangan Rekayasa Perangkat Lunak 9](#_TOC_250054)
     6. Model SDLC (Software Development Life Cycle) 9
        1. [Model Waterfall 10](#_TOC_250053)
        2. [Model Prototipe 11](#_TOC_250052)
        3. Model RAD (*Rapid Application Development*) 13
        4. [Model Interative 16](#_TOC_250051)
        5. [Model Spiral 17](#_TOC_250050)
  2. [Unified Modeling Language 18](#_TOC_250049)
     1. Defenisi Unified Modeling Language 19
     2. [Sejarah UML 19](#_TOC_250048)
     3. [Diagram UML 19](#_TOC_250047)
        1. [Use Case Diagram 20](#_TOC_250046)
        2. [Class Diagram 25](#_TOC_250045)
        3. [Sequence Diagram 27](#_TOC_250044)
        4. [Collaboration Diagram 30](#_TOC_250043)
        5. [Statechart Diagram 31](#_TOC_250042)
        6. [Activity Diagram 32](#_TOC_250041)
        7. Development Diagram 34
  3. Kecerdasan Buatan 35
     1. Defenisi Kecerdasan Buatan 36
     2. Sejarah Kecerdasan Buatan 36
     3. Kecerdasan Alami dan Kecerdasan Buatan 38
  4. Sistem Pakar 39
     1. Defenisi Sistem Pakar 40
     2. Konsep dasar Sistem Pakar 40
        1. [Kepakaran 40](#_TOC_250040)
        2. [Aturan – Aturan (*Rule*) 41](#_TOC_250039)
        3. Pakar (*Expert*) 41
        4. Inferensi (*Interferencing*) 42
        5. Kemampuan Menjelasakan(*Explanation capability*) 42
        6. Pemindahan Kepakaran (*Transfering Expertise*) 44
        7. [Ciri – Ciri dan Area Permasalahan Sistem Pakar 44](#_TOC_250038)
     3. Struktur Sistem Pakar 46
     4. Kelebihan dan Kekurangan Sistem Pakar 49
     5. Keuntungan Sistem Pakar 51
  5. Naïve Bayes 51
     1. Defenisi Naïve Bayes 51
     2. Rumus yang digunakan pada Naïve Bayes 52
  6. Website 53
     1. Defenisi Website 53
     2. Jenis – Jenis Website 54
  7. Webserver dan Webbrowser 55
  8. Pemrograman Web 55
     1. PHP (Hypertext Prepocessor) 55
     2. Keungulan PHP 56
     3. Database 56
     4. MySQL 57
  9. Gonore 59
     1. Defenisi Gonore 59
     2. Jenis – Jenis Penyakit Gonore 60
     3. Gejala – Gejala yang Ditimbulkan Penyakit Gonore 61

[BAB III METODOLOGI PENELITIAN 64](#_TOC_250037)

* 1. [Kerangka Penelitian 64](#_TOC_250036)
  2. [Tahapan Penelitian 64](#_TOC_250035)
     1. [Penelitian Pendahuluan 65](#_TOC_250034)
     2. [Pengumpulan Data 65](#_TOC_250033)
     3. [Analisa 69](#_TOC_250032)
     4. [Perancangan 69](#_TOC_250031)
     5. [Implementasi 72](#_TOC_250030)
     6. [Pengujian 72](#_TOC_250029)

BAB IV ANALISA DAN PERANCANGAN 74

* 1. [Anggaran Biaya 74](#_TOC_250028)
  2. Jadwal Penelitian  [89](#_TOC_250020)

DAFTAR PUSTAKA 81

Lampiran 1.Biodata Ketua dan Anggota Peneliti83

**RINGKASAN**

Sistem pakar adalah sistem yang berusaha mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti layaknya seorang pakar. Dengan sistem pakar ini, masyarakat dapat memperoleh suatu informasi atau dapat menyelesaikan permasalahan yang dihadapi yang sebenarnya hanya dapat diperoleh dengan bantuan para ahli dibidangnya. Dalam penelitian ini dijelaskan cara sistem pakar mendiagnosa penyakit *Gonore* dengan menggunakan metode *Naïve Bayes*. Dengan menggunakan metode *Naïve Bayes*, dapat memprediksi nilai probabiltas dimasa depan berdasarkan pengalaman dimasa sebelumnya yang di alami penderita penyakit *Gonore*. Program sistem pakar ini diimplementasikan dengan menggunakan bahasa pemrograman PHP dan database MySQL berbasis *Web*. Kesimpulan dari sistem ini mudah untuk dikembangkan dan mudah juga dalam menggunakannya, dan berguna bagi masyarakat yang ingin mengetahui tentang penyakit *Gonore*.

**Kata Kunci: Sistem Pakar, *Gonore*, *Naïve Bayes*, *Web*.**

# BAB I PENDAHULUAN

* 1. **Latar Belakang Masalah**

Sistem pakar adalah perangkat lunak komputer yang menggunakan pengetahuan (aturan-aturan tentang sifat dari unsur suatu masalah), fakta dan teknik inferensi untuk masalah yang biasanya membutuhkan kemampuan seorang ahli. Sistem pakar adalah suatu sistem komputer yang menyamai kemampuan pengambilan keputusan dari seorang pakar. Prof. Edward Fieganbaum (1982 : 1) dari universitas Stanford sebagai seorang pelopor awal dari teknologi sistem pakar, mendefenisikan sistem pakar sebagai suatu program komputer cerdas yang menggunakan *knowledge* (pengetahuan) dan prosedur inferensi untuk menyelesaikan masalah yang cukup sulit sehingga membutuhkan seorang ahli untuk menyelesaikan. Tujuan utama sistem pakar bukan untuk menggantikan kedudukan seorang ahli atau seorang pakar, tetapi hanya untuk memasyarakatan pengetahuan dan pengalaman dari pakar.

*Gonore* dalam arti luas mencakup semua penyakit yang disebabkan oleh bakteri. Menurut Jawas (dalam Nurdin *et al.*, 2017) *gonorrhoeae* merupakan jenis penyakit menular seksual (PMS) klasik yang disebabkan oleh infeksi bakteri *Neisseria gonorrhoeae* yang mempunyai insidens yang tinggi diantara infeksi penyakit menular seksual (PMS). Infeksi ini terjadi secara luas di seluruh dunia dengan prevelensi yang lebih tinggi di berbagai negara berkembang termasuk Indonesia. Infeksi penyakit *Gonore* dapat menyerang pria maupun wanita, dimana terdapat beberapa jenis penyakitnya yaitu, Uretritis Gonore, Servisitis Gonore, Faringitis Gonore, Prakitis Gonore. IMS (infeksi menular seksual) penyebabnya

lebih dari 30 bakteri, virus, parasite, jamur, dan berbagai jenis yang mana penyebarannya melalui kontak seksual dan secara umum infeksi ini memiliki karakteristik asimtomatik atau tidak menunjukan gejalanya sama sekali (Puspita, 2017). Sebagian besar penderita penyakit *Gonore* pada kelamin atau organ intim malu untuk berkonsultasi dengan dokter atau pakar secara langsung. Dengan melakukan diagnosa, diharapkan dapat mencari informasi lebih mendalam tentang suatu penyakit, khusunya penyakit *Gonore* tersebut. Selain itu, bisa menemukan gejala dan pencegahan dari penyakit tersebut berdasarkan pengetahuan yang diberikan oleh dokter yang menanganinya.

Dalam penelitian ini dikembangkan sistem pakar untuk diagnosa penyakit *Gonore* dengan menggunakan metode *Naïve Bayes*. Metode ini merupakan klasifikasi probabilitas sederhana berdasarkan pada teorema bayes. Keuntungan dari klasifikasi adalah bahwa ia hanya membutuhkan sejumlah kecil data pelatihan untuk memperkirakan parameter (sarana dan varians dari variable) yang diperlukan untuk klasifikasi (Puspa, 2018). Perancangan *Naïve Bayes* ini bisa digunakan untuk membantu pakar dalam mengkalsifikasi penyakit dan memberikan cara penanggulangannya. Metode *Naïve Bayes* dipilih karena metode ini memperhatikan seluruh fitur pada data latih sehingga membuat metode ini optimal dalam melakukan proses klasifikasi dan memiliki tingkat akurasi yang cukup tinggi dibanding dengan metode lainnya (Prayoga, Hidayat and Dewi, 2018).

Berdasarkan penjelasan yang ada diatas maka penulis mencoba mengambil kesimpulan judul “PERANCANGAN SISTEM PAKAR UNTUK DIAGNOSA PENYAKIT GONORE DENGAN MENGGUNAKAN METODE NAÏVE BAYES BERBASIS WEB”.

# Tujuan Penelitian

Adapaun manfaat yang diharapkan dalam penelitian ini adalah sebagai berikut :

* + 1. Membuat sistem pakar berbasis web yang dapat digunakan untuk mendiagnosa penyakit gonore.
    2. Menerapkan metode *naïve bayes* untuk menangani penyakit dan memberi solusi pada diagnosa penyakit gonore.
    3. Menghasilkan sistem pakar yang dapat mendiagnosa penyakit gonore.

# Manfaat Penelitian

Adapaun manfaat yang diharapkan dalam penelitian ini adalah sebagai berikut :

* + 1. Membantu masyarakat untuk mendapatkan informasi dan menemukan solusi terhadapa penyakit gonore.
    2. Mempermudah masyarakat untuk melakukan konsultasi dengan sistem layaknya berkonsultasi dengan seorang pakar untuk mengetahui apakah masyarakat tersebut terdiagnosa penyakit gonore atau tidak terdiagnosa.
    3. Masyarakat menjadi lebih peduli terhadap kesehatannya dan kebersihan terutama pada alat vital.

# Batasan Masalah

Agar penelitian ini terarah dan tujuan dari penulis ini tercapai sesuai diharapkan, maka perlu adanya batasan masalah, yaitu :

* + 1. Metode naïve bayes digunakan untuk menghitung peluang penyakit gonore berdasarkan gejala-gejala yang timbul dan nilai yang diberikan oleh seorang pakar.
    2. Sistem pakar ini dirancang menggunakan bahasa pemrograman PHP dan MySQL sebagai basis data dan metode yang digunakan adalah *naïve bayes*.
    3. Penulis hanya membahas beberapa jenis penyakit gonore yaitu, Uretritis Gonore, Faringitis Gonore, Prokitis Gonore, Servisitis Gonore.

# BAB II LANDASAN TEORI

# Rekayasa Perangkat Lunak

Rekayasa Perangkat Lunak (*Software Engineering*) adalah bidang profesi yang mendalami cara-cara pengembangan perangkat lunak termasuk pembuatan, pemeliharaan, manajemen organisasi pengembangan perangkat lunak, dan sebagainya (Janner, 2010).

# Defenisi Perangkat Lunak

Menurut (Shalahuddin and Rosa, 2018), Perangkat lunak (*Software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*).

Rekayasa atau teknik penerapan ilmu dan teknologi untuk menyelesaikan permasalahan manusia hal ini diselesikan lewat pengetahuan, matematika, dan pengalaman praktis yang diterapkan untuk mendesain objek atau proses yang berguna (Janner, 2010).

Pada tahun 2004, istilah rekayasa perangkat lunak secara umum digunakan dalam tiga arti yaitu (Janner, 2010):

* + - 1. Sebagai istilah umum untuk berbagai kegiatan yang dulunya bernama pemograman atau analisis system.
      2. Sebagai istilah yang luas untuk analisis teknik dari semua aspek – aspek praktis yang bertentangan dengan teori pemograman komputer, dan
      3. Sebagai istilah yang mewujudkan advokasi suatu pendekatan spesifik ke programan computer, suatu hal yang mendesak yang diperlukan sebagai profesi rekayasa dari pada sebuah seni atau kerajinan, dan advokasi dari konfikasi praktis yang disarankan dalam bentuk metodologi rekayasa perangkat lunak.

# Defenisi Rekayasa Perangkat Lunak

Rekayasa perangkat lunak (*Software Engineering*) merupakan pembangunan dengan menggunakan prinsip atau konsep rekayasa dengan tujuan menghasilkan perangkat lunak yang bernilai ekonomi yang dipercaya dan bekerja secara efisien dengan menggunakan mesin. Oleh karena itu, Rekayasa perangkat lunak dibutuhkan agar perangkat lunak yang dibuat tidak hanya menjadi perangkat lunak yang tidak terpakai (Shalahuddin and Rosa, 2018).

# Proses Perangkat Lunak

Selama pengembangan suatu perangkat lunak, maka haru dilakukan proses rekayasa perangkat lunak. Secara garis besar, proses rekayasa perangkat lunak dapat digambarkan sebagai berikut :

Mulai

Analisis

Pengujian

Perancang



Implemen

Sumber : Shalahuddin and Rosa, 2018

# Gambar 2.1 Tahapan Umum Rekayasa Perangkat Lunak

Proses perangkat lunak (*software process*) adalah sekumpulan aktifitas yang memilki tujuan untuk mengembangkan atau mengubah perangkat lunak. Secara umum ada beberapa proses perangkat lunak yang terdiri dari (Shalahuddin and Rosa, 2018):

* + - 1. Pengumpulan (*Spesifikasi*)

Mengetahui apa saja yang harus dapat dikerjakan sistem perangkat lunak dan batasan pengembangan perangkat lunak.

* + - 1. Pengembangan (*Development*)

Pengembangan perangkat lunak untuk menghasilkan sistem perangkat lunak.

* + - 1. Validasi (*Validation*)

Memeriksa apakah perangkat lunak sudah memenuhi keinginan pelanggan (*customer*).

* + - 1. Evolusi (*Evolution*)

Mengubah perangkat lunak untuk memenuhi perubahan kebutuhan pelanggan (*customer*).

# Kriteria Perangkat Lunak

Rekayasa perangkat lunak lebih fokus pada bagaimana membuat perangkat lunak yang memenuhi kriteria berikut (Shalahuddin and Rosa, 2018):

* + - 1. Dapat terus dipelihara setelah perangkat lunak selesai dibuat seiring berkembangnya teknologi dan lingkungan (maintainability).
      2. dapat diandalkan dengan proses bisnis yang dijalankan dan perubahan yang terjadi (dependability dan robust).
      3. Efisien dari segi sumber daya dan pengguna.
      4. Kemampuan untuk dipakai sesuai dengan kebutuhan (usability).

Dari kriteria di atas maka perangkat lunak yang baik adalah perangkat lunak yang memenuhi kebutuhan pelanggan. *Costumer* atau *user* (pemakai perangkat lunak) atau berorientasi pada pelanggan atau pemakai perangkat luna, bukan berorientasi pada pembuat atau pengembang peragkat lunak.

# Tantangan Rekayasa Perangkat Lunak

Menurut (Shalahuddin and Rosa, 2018) menyatakan tantangan yang dihadapi dalam proses rekayasa perangkat lunak yaitu sebagai berikut:

* + - 1. Tantangan warisan dimana perngkat lunak dikembangkan selama bertahun-tahun oleh orang-orang yang berbeda, hal ini dapat menyebabkan ketidakpahaman atau perubahan tujuan pembuatan perangkat lunak.
      2. Tantangan *heterogenitas*, dimana perangkat lunak harus dapat beradaptasi dengan teknologi yang terus berkembang dengan semakin luasnya lingkungan distribusi perangkat lunak
      3. Tantangan pengiriman bahwa perangkat lunak dengan sekala besar dan kompleks sekalipun dapat sampai ke tangan pelanggan (*customer*) atau *user* dengan cepat dan kualitas tetap terjaga.
    1. **Model SDLC (*Software Development Language Cycle*)**

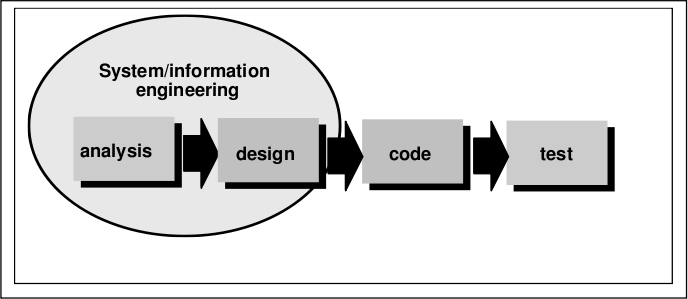
SDLC atau *Software Development Life Cycle* atau sering disebut *System Development Life Cycle* adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodelogi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya (berdasarkan *best practice* atau cara-cara yang sudah teruji baik (Shalahuddin and Rosa, 2018).

SDLC memiliki beberapa model dalam penerapan tahapan prosesnya.

Model-model SDLC sebagai berikut (Shalahuddin and Rosa, 2018):

# Model Waterfall

Model penerapan tahapan air mancur proses (*waterfall model*) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut mulai dari tahap analisa, desain, pengkodean, pengujian, hingga ke tahap maintenance. Berikut ini adalah gambar model air terjun:



Sumber : Shalahuddin and Rosa, 2018

# Gambar 2.2 Model Waterfall

1. Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk mespesifikasikan kebutuhan perangkat lunak agar dapat dipahami *user*.

1. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data,

arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean.

1. Pembuatan kode program

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

1. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi lojik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan *(error)* dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

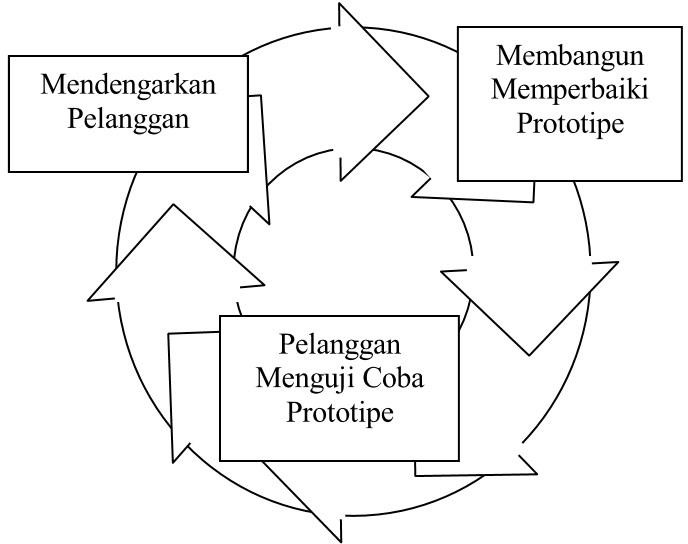
# Model Prototipe

Sering pelanggan (*customer*) membayangkan kumpulan kebutuhan yang diinginkan tapi tidak terspesifikasi secara detail dari segi masukan (*input*), proses, maupun keluaran (*output*). Disisi lain seorang pengembang perangkat lunak harus merepresentasikan sebuah kebutuhan secara detail dari segi teknis dimana pelanggan sering kurang mengerti mengenai hal teknis ini (Shalahuddin and Rosa, 2018).

Model prototipe (*prototyping model*) dimulai dari mengumpulkan kebutuhan pelanggan terhadapa perangkat lunak yang akan dibuat. Program prototype biasanya merupakan program yang belum jadi. Program ini biasanya menyediakan tampilan dengan simulasi alur perangkat lunak sehingga tampak seperti perangkat lunak yang sudah jadi. Program prototype ini dievaluasi oleh pelanggan atau *user* sampai

ditemukan spesifikasi yang sesuai dengan keinginan pelanggan atau *user.*

Berikut adalah gambar dari model prototipe:



Sumber : Shalahuddin and Rosa, 2018

# Gambar 2.3 Model Prototipe

*Mock-up* adalah sesuatu yang digunakan sebagai model desain yang digunakan untuk mengajar, demonstrasi, evaluasi desain, promosi atau keperluan lain. Sebuah *mock-up* disebut sebagai prototipe perangkat lunak jika menyediakan atau mampu mendemostrasikan sebagian besar fungsi sistem perangkat lunak dan memungkinkan pengujian desain sistem perangkat lunak. Iterasi terjadi pada pembuatan prototipe sampai sesaui dengan keinginan pelanggan (*customer*) atau *user* (Shalahuddin and Rosa, 2018).

Model prototipe mempunyai kelemahan sebagai berikut (Shalahuddin and Rosa, 2018):

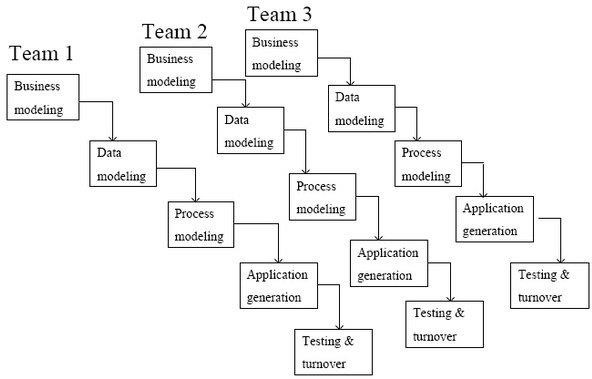
* + - * 1. Pelanggan dapat sering mengubah-ubah atau menambah- tambah spesifikasi kebutuhan karena menganggap aplikasi

sudah dengan cepat dikembangan, karena adanya iterasi ini dapat menyebabkan pengembang banyak mengalah dengan pelanggan karena perubahan atau penambahan spesifikasi kebutuhan perangkat lunak.

* + - * 1. Pengembangan lebih sering mengambil kompromi dengan pelanggan untuk mendapatkan *prototipe* dengan waktu yang cepat sehingga pengembang lebih sering melakukan segala cara guna menghasilkan *prototipe* untuk didemontrasikan.
      1. **Model RAD (*Rapid Application Development*)**

*Rapid Application Development* (*RAD*) adalah model proses pengembangan perangkat lunak yang bersifat incremental terutama untuk waktu pengerjaan yang pendek. Model RAD adalah adaptasi dari model air terjun versi kecepatan tinggi dengan menggunakan model air terjun untuk pengembangan setiap komponen perangkat lunak (Shalahuddin and Rosa, 2018).

Model RAD membagi tim pengembang menjadi beberapa untuk mengerjakan beberapa komponen masing-masing pengerjaan dapat dilakukan secara paralel. Berikut adalah gambar dari model RAD:



Sumber : Shalahuddin and Rosa, 2018

# Gambar 2.4 Model RAD

* + - * 1. Pemodelan bisnis

Pemodelan yang dilakukan untuk memodelkan fungsi bisnis untuk mengetahui informasi apa yang terkait proses bisnis, informasi apa saja yang harus dibuat, siapa yang harus membuat informasi itu, bagaimana alur informasi itu, proses apa saja yamg terkait informasi itu.

* + - * 1. Pemodelan data

Memodelkan data apa saja yang dibutuhkan berdasarkan pemodelan bisnis dan mendefenisikan atribut-atributnya beserta relasinya dengan data-data yang lain.

* + - * 1. Pemodelan proses

Mengimplementasikan fungsi bisnis yang sudah didefenisikan terkait denagn pendefenisian data.

* + - * 1. Pembuatan aplikasi

Mengimplemtasikan pemodelan proses dan data menjadi program. Model RAD sangat menganjurkan pemakaian komponen yang sudah ada jika dimungkinkan.

* + - * 1. Pengujian dan pergantian

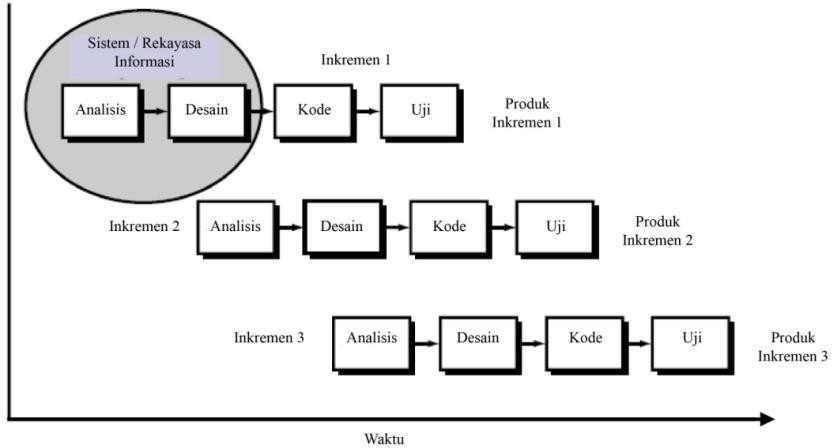
Menguji komponen-komponen yang dibuat. Jika sudah teruji maka tim pengembang komponen dapat beranjak untuk mengembangkan komponen berikutnya.

Model RAD memiliki kelemahan sebagai berikut (Shalahuddin and Rosa, 2018):

1. Untuk pembuatan sistem perangkat lunak dengan skala besar maka model RAD akan memerlukan sumber daya manusia yang cukup besar untuk membentuk tim yang mengembangkan komponen-komponennya.
2. Jika tidak ada persetujuan untuk mengembangkan perangkat lunak secara cepat (*rapid*) maka proyek dengan model ini akan gagal karena hanya akan bingung mendefinisikan kebutuhan pelanggan (*customer*) atau *user*.
3. Jika sistem perangkat lunak yang akan dibuat tidak bisa dimodulkan (dibagi-bagi menjadi beberapa komponen) maka model RAD tidak dapat digunakan untuk membuat sistem perangkat lunak ini karena terlalu banyak campur tangan antara tim.
4. Model RAD tidak cocok digunakan untuk sistem perangkat lunak yang memiliki resiko teknis sangat tinggi.

# Model Interative

iterative (*interative model*) mengkombinasikan proses-proses pada model air terjun dan iteratif pada model prototipe. Model inkremental akan menghasilkan versi-versi perangkat lunak yang sudah mengalami penambahan fungsi untuk setiap pertambahannya (*inkremen/increment*). Berikut adalah gambar dari model incremental:



Sumber : Shalahuddin and Rosa, 2018

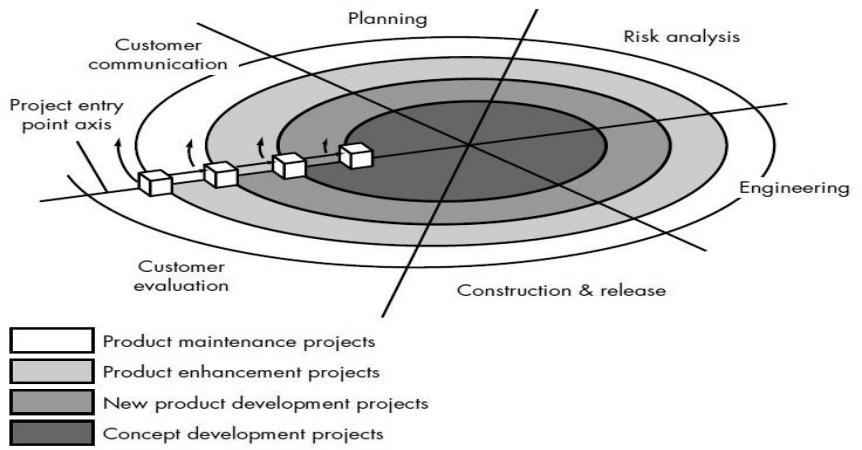
# Gambar 2.5 Model Intertif

Model incremental dibuat untuk mengatasi kelemahan dari model air terjun yang tidak mengakomodasi iterasi, dan mengatasi kelemahan dari metode prototipe yang memiliki proses terlalu pendek dan setiap iteratif prosesnya tidak selalu menghasilkan produk (bisa jadi hanya prototipe). Model inkremental menghasilkan produk/aplikasi untuk setiap tahapan inkremen. Model inkremental sangat cocok digunakan jika staf

tidak dapat memiliki pergantian (*turnover*) yang tinggi sehingga staf tidak dapat terus ikut dalam pengembangan perangkat lunak.

# Model Spiral

Model spiral (*spiral model*) memasangkan iteratif pada model prototipe dengan kontrol dan aspek sistematik yang diambil dari model air terjun. Model spiral menyediakan pengembangan dengan cara cepat dengan perangkat lunak yang memiliki versi yang terus bertambah fungsinya (*increment*). Berikut adalah gambar dari model spiral:



Sumber : Shalahuddin and Rosa, 2018

# Gambar 2.6 Model Spiral

Pada iterasi awalnya maka yang dihasilkan adalah prototipe sedangkan pada iterasi akhir yang dihasilkan adalah perangkat lunak yang sudah lengkap. Model spiral dibagi menjadi beberapa kerangka aktifitas atau disebut juga wilayah kerja (*task region*) sebagai berikut:

* + - * 1. Komunikasi dengan pelanggan (*customer communication*) Aktifitas ini diperlukan untuk membangun komunikasi efektif

antara pengembang (*developer*) dan pelanggan (*customer*).

* + - * 1. Perancangan (*planning*)

Aktifitas ini diperlukan untuk mendefinisikan sumber daya, waktu, dan informasi yang terkait dengan proyek.

* + - * 1. Analisis risiko (*risk analisys*)

Aktifitas ini diperlukan untuk memperkirakan risiko dari segi teknis maupun manajemen.

* + - * 1. Rekayasa (*engineering*)

Aktifitas ini diperlukan untuk membangun salah satu atau lebih representasi dari aplikasi perangkat lunak (dapat juga berupa prototipe).

* + - * 1. Konstruksi dan peluncuran (*construction and release*) Aktifitas ini diperlukan untuk mengkonstruksi, menguji, melakukan instalasi, dan menyediakan dukungan terhadap *user* (misalnya dari segi dokumentasi dan pelatihan).
        2. Evaluasi pelanggan (*customer evaluation*)

Aktifitas evaluasi ini diperlukan untuk mendapatkan umpan balik

berdasarkan evaluasi representasi perangkat lunak yang dihasilkan dari proses rerkayasa dan diimplementasikan pada tahap instalasi.

## Unified Modeling Language

Menurut (Anisah and Kuswaya, 2017), *Unified Modeling Language* (UML) merupakan suatu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem software yang terkait dengan objek.

* + 1. **Defenisi *Unified Modeling Language***

*Unified Modeling Language* Salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek (Wira, Putra and Andriani, 2019).

# Sejarah UML

Bahasa pemograman beorientasi objek yang pertama dikembangkan dikenal dengan nama Simula-67 yang dikembangkan pada tahun 1967. Bahasa pemograman ini kurang berkembang dan dikembangkan lebih lanjut, namun dengan kemunculannya telah memberikan sumbanga yang besar pada developer pengembang bahasa pemograman berorientasi objek selanjutnya (Shalahuddin and Rosa, 2018).

Perkembangan aktif dari pemograman berorientasi objek mulai menggeliat ketika berkembangnya bahasa pemograman Smalltalk pada awal tahun 1980-an yang kemudian diikuti dengan berkembangnya bahasa pemogrmaan berorientasi objek yang lainnya seperti C objek, C++, Eiffel, dan CLOS. Pada 1996, *Object Management Group* (OMG) mengajukan proposal agar adanya standarisasi pemodelan berorientasi objek dan pada bualn September 1997 UML diakomodosikan oleh OMG sehingga sampai saaat ini UML tealh memberikan kontribusi yang cukup besar didalam metodologi berorientasi objek dan hal – hal yang terkait didalamnya (Shalahuddin and Rosa, 2018).

# Diagram UML

*Unified Modeling Lamguage* (UML) mendefenisikan diagram-diagram sebagai *Use case diagram*, *Class diagram*, *Sequence diagram*, *Collaboration*

*diagram, Statechart diagram*, *Activity diagram*, *Deployment diagram* (Suendri, 2019).

## Use Case Diagram

*Use case* merupakan pemodelan untuk kelakuan sistem informasi yang akan dibuat. Use case bekerja dengan mendeskripsikan tipikal interaksi antara user sebuah sistem dengan sistemnnya sendiri melalui sebuah cerita bagaimana sistem itu dipakai (Wira, Putra and Andriani, 2019).

Syarat penamaan pada *use case* adalah ama didefenisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utam pada *use case* yaitu pendefenisian apa yang disebut aktor dan *use case* (Shalahuddin and Rosa, 2018).

* + - * 1. Aktot merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walapun simbol dari aktor adalah gambar orang, tapi aktor belum tentumerupakan orang.
        2. Use case merupakan fungsionalitasyang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

Berikut adalah simbol-simbol yang ada pada diagram *use case* :

**Tabel 2.1 Simbol - Simbol *Use Case Diagram***

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Simbol** | **Nama Simbol** | **Deskripsi** |
| 1 |  | *Use case* | Fungsionalitas yang disediakan  sistem sebagai unit-unit yang |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja  diawal frase name *use case* |
| 2 |  | Aktor / *actor* | Orang, proses atau sistem lain yang berintekrasi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan dengan menggunakan kata benda di awal frase nama  aktor. |
| 3 |  | Asosiasi /  *assosiation* | Komunikasi antar aktor dan *use case* yang berpartisipasi pada *use case* atau *use case* yang memiliki  interaksi dengan aktor. |
| 4 | *<<extend>>* | Ekstensi /  *extend* | Relasi *use case* tambahan ke sebuah *use case* dimana *use case* yang ditambahkan dapat berdiri sendiri walau tanpa *use case* tambahan itu; mirip dengan prisip  *inheritance* pada pemrograman |



|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | berorientasi objek; biasanya *use case* tambahan memiliki nama depan yang sama dengan *use case* yang ditambahkan, misal.  **Validasi**  .  <<extend>>  **Validasi user**  <<extend>>  **Validasi sidik jari**  Arah panah mengarah pada *use case* yang ditambahkan; biasanya *use case* yang menjadi *extend*-nya merupakan jenis yang sama dengan *use case* yang menjadi  induknya. |
| 5 |  | Generalisasi / generalization | Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah *use case* dimana fungsi yang satu adalah fungsi yang lebih umum dari yang lainnya, misalnya : |



|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | **Ubah data**  **Mengelola data**  **Hapus data**  Arah panah mengarah pada *use case* yang menjadi generalisasinya (umum). |
| 6 | <<include>>    << *uses* >> | Menggunakan  / *include / uses* | Relasi *use case* tambahan ke sebuah *use case* dimana *use case* yang ditambahkan memerlukan *use case* ini untuk menjalankan fungsinya atau sebagai syarat dijalankan *use case* ini.  Ada dua sudut pandang yang cukup besar mengenai include di  *use case*: |
|  |  |  |  |
|  |  |  | 1. *Include* berarti *use case* yang ditambahkan akan selalu dipanggil saat *use case* tambahan dijalankan, misal pada kasus berikut: |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | **Validasi**  **username**  <<include>>  **login**  2. Include berarti *use case* yang tambahan akan selalu melakukan pengecekan apakah *use case* yang ditambahkan telah dijalankan sebelum *use case* tambahan dijalankan, missal pada kasus berikut :  **Validasi**  <<*include*>>  **Ubah data**  Kedua interpretasi di atas dapat dianut salah satu atau keduanya tergantung pada pertimbangan dan interpretasi yang dibutuhkan. |

Sumber : Shalahuddin and Rosa, 2018

## Class Diagram

Kelas (Class) adalah sebuah spesifikasi yang jika diinstansikan akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan perancangan berorientasi objek. Kelas menggambarkan keadaan (atribut/properti) dari suatu sistem, sekaligus manawarkan layanan untuk memanipulasi keadaan tersebut dengan menggunakan (metode/fungsi) (Tohari, 2014).

Kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi susuai dengan kebutuhan sistem sehingga pembuat perangkat lunak atau *programmer* dapat membuat kelas-kelas didalam program perangkat lunak sesuai dengan perancangan diagram kelas. Susunan struktur kelas baik pada program kelas sebaiknya memiliki jenis- jenis kelas berikut (Shalahuddin and Rosa, 2018):

* + - * 1. Kelas main

kelas yang mempunyai fungsi awal dieksekusi ketika sistem dijalankan.

* + - * 1. Kelas yang menangani tampilan sistem (*view*)

kelas yang medefinisikan dan mengatur tampilan ke pemakai.

* + - * 1. Kelas yang diambil dari pendefenisisan *use case* (*controller*) kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian use case.
        2. Kelas yang diambil dari pendefenisisan data (*model*)

kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data.

Berikut adalah simbol-simbol yang ada pada diagram kelas:

**Tabel 2.2 Simbol - Simbol *Class Diagram***

|  |
| --- |
| **nama\_kls** |
| +atribut |
| +operasi() |

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Simbol** | **Nama Simbol** | **Deskripsi** |
| 1 |  | Kelas | Kelas pada struktur sistem. |
| 2 |  | Antarmuka/  *Interface* | Sama dengan konsep interface dalam pemograman berorientasi objek. |
| 3 |  | Asosiasi /  *assosiation* | Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan *multiplicity*. |
| 4 |  | Asosiasi berarah/ *Directed Association* | Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasnya juga disertai dengan *multiplicity.* |
| 5 |  | *Generalisasi* | Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus). |

|  |  |  |  |
| --- | --- | --- | --- |
| 6 |  | Kebergantungan  / *Dependency* | Relasi antar kelas dengan makna kebergantungan antar kelas. |
| 7 |  | Agregasi / Aggregation | Relasi antar kelas dengan makna semua bagian (*whole-part*). |

Sumber : Shalahuddin and Rosa, 2018

## Sequence Diagram

*Sequence diagram* menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Gambaran sequence diagram dibuat minimal sebnayak pendefinisan use case yang memiliki proses sendiri atau yang penting semua use case yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada sequence diagram sehingga semakin banyak use case yang didefinisikan, maka sequence diagram yang harus dibuat juga semakin banyak (Wira, Putra and Andriani, 2019).

Berikut adalah simbol-simbol yang ada pada diagram sequence:

**Tabel 2.3 Simbol - Simbol *Sequence Diagram***

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Simbol** | **Nama**  **Simbol** | **Deskripsi** |
| 1 | Nama aktor | Aktor | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem  informasi yang akan dibuat itu |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | |  | sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor. |
| 2 | *|*  *|*  *|*  *|* | | Garis hidup /  *lifeline* | Menyatakan kehidupan suatu objek. |
| 3 |  | | Objek | Menyatakan objek yang berinteraksi pesan |
| Nama objek : nama kelas |  |
|  | |
| 4 |  | | Waktu aktif | Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya, misalnya  **2:cekStatusLogin()**  **1: login 3: open()**  Maka cekStatusLogin() dan open() dilakukan di dalam metode login() Aktor tidak memiliki waktu aktif |

|  |  |  |  |
| --- | --- | --- | --- |
| 5 | <<create>> | Pesan tipe create | Menyatakan suatu objek membuat objek yang lain, arah panah  mengarah pada objek yang dibuat. |
| 6 |  |  | Menyatakan suatu objek memanggil operasi/ metode yang ada pada objek lain atau dirinya sendiri, |
|  | 1. nama  \_metode() | Pesan tipe call | 1:nama\_metode() Arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas  objek yang berinteraksi. |
| 7 | 1: masukan | Pesan tipe send | Menyatakan bahwa suatu objek mengirim data/masukan/informasi ke objek lainnya, arah panah  mengarah pada objek yang dikirimi. |
| 8 | 1: keluaran | Pesan tipe return | Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu  kembalian ke objek tertentu, arah |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | panah mengarah pada objek yang  menerima kembalian. |
| 9 | <<destroy>> | Pesan tipe destroy | Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah psds objek yang diakhiri, sebaiknya jika ada create maka ada  destroy |

Sumber : Shalahuddin and Rosa, 2018

## Collaboration Diagram

*Collaboration Diagram* merupakan cara alternative untuk menggambarkan scenario dari suatu sistem. Diagram ini menggambarkan interaksi objek yang diatur oleh objek sekelilingnya dan hubungan antara setiap objek dengan objek lainnya (Tohari, 2014).

Berikut adalah simbol-simbol yang ada pada diagram kolaborasi:

**Tabel 2.4 Simbol - Simbol *Collaboration Diagram***

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Simbol** | **Nama Simbol** | **Deskripsi** |
| 1 | Nama\_objek : nama\_kelas | Objek | Objek yang melakukan interaksi pesan. |
| 2 |  | *Link* | Relasi antara objek yang  menghubngkan objek satu dengan |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | lainya atau dengan diri sendiri.  Nama\_objek : nama\_kelas |
| 3 |  | Arah pesan /  *stimulus* | Arah pesan yang terjadi, jika pada suatu *link* ada dua arah pesan yang berbeda maka arah juga digambarkan du arah pada dua sisi *link*. |

Sumber : Shalahuddin and Rosa, 2018

## Statechart Diagram

*State machine diagram* atau *statechart diagram* atau dalam bahasa indonesia disebut diagram mesin status atau sering juga disebut diagram status digunakan untuk menggambarkan perubahan status atau transisi status dari sebuah mesin atau sistem atau objek. *State machine diagram* merupakan pengembangan dari diagram *Finite State Automata* dengan penambahan fitur dan konsep yang baru (Shalahuddin and Rosa, 2018).

Berikut adalah simbol-simbol yang ada pada *state machine diagram*:

**Tabel 2.5 Simbol - Simbol *Statechart Diagram***

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Simbol** | **Nama Simbol** | **Deskripsi** |
| 1 |  | *Start/*Status Awal  *(Initial State)* | Keadaan awal pada saat system mulai hidup. |

|  |  |  |  |
| --- | --- | --- | --- |
| 2 |  | *End /* Status Akhir (*Final State*) | Keadaan akhir dari daur hidup suatu system. |
| 3 | *Event* | *Event* | Kegiatan yang menyebabkan berubahnya status mesin. |
| 4 | state | *State* | Sistem pada waktu tertentu. *State* dapat berubah jika ada *event* tertentu yang memicu perubahan tersebut. |

Sumber : Shalahuddin and Rosa, 2018

## Activity Diagram

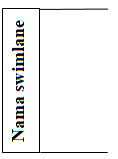
*Activity* diagram memodelkan *workflow* proses bisnis dan urutan aktivitas dalam sebuah proses. Diagram ini sangat mirip dengan *flowchart* karena memodelkan *workflow* dari suatu aktivitas ke aktivitas lainnya atau dari aktivitas ke status. Membuat *activity* diagram pada awal pemodelan proses cukup menguntungkan untuk membantu memahami keseluruhan proses. *Activity* diagram juga bermanfaat untuk menggambarkan *parallel behavior* atau menggambarkan interaksi antara beberapa *use case* (Tohari, 2014).

Berikut adalah simbol-simbol yang ada pada diagram aktifitas:

**Tabel 2.6 Simbol - simbol pada *Activity Diagram***

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Simbol** | **Nama Simbol** | **Deskripsi** |
| 1 |  | Status Awal | Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal. |
| 2 | aktivitas | Aktivitas | Aktivitas yang dilakukan sistem, aktivitas biasaya diawali dengan kata kerja. |
| 3 |  | Percabangan/  *Decision* | Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu. |
| 4 |  | Penggabungan / join | Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu. |
| 5 |  | Status akhir | Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir. |

|  |  |  |  |
| --- | --- | --- | --- |
| 6 | **Nama**  **swimlane**  Atau | Swimlane | Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi. |

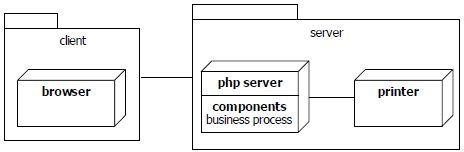
Sumber : Shalahuddin and Rosa, 2018

## Deployment Diagram

*Deployment diagram* digunakan untuk menggambarkan detail bagaimana komponen disusun di infrastruktur sistem (Hendini, 2016). Diagram *deployment* juga dapat digunakan untuk memodelkan hal-hal berikut:

Sistem tambahan (*embedded system*) yang menggambarkan rancangan *devide*, *node*, *hardware*.

Sistem client / server misalnya seperti gambar berikut:



Sumber : S, Rosa A. dan M. Shalahuddin, 2018

**Gambar 2.7 Diagram *Deployment* Sistem Client / Server**

Berikut adalah simbol-simbol yang ada pada diagram *deployment*:

**Tabel 2.7 Simbol - Simbol *Deployment Diagram***

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Simbol** | **Nama Simbol** | **Deskripsi** |
| 1 |  | *Package* | *Package* merupakan sebuah bungkusan dari satu atau lebih *node* |
| 2 |  | *Node* | Node adalah sumber daya fisik yang menjalankan kode komponen. |
| 3 |  | *Dependency* | Ketergantungan antara *node*, arah  panah mengarah pada *node* yang dipakai. |
| 4 |  | *Link* | Relasi antara *node.* |

Sumber : Shalahuddin and Rosa, 2018

# Kecerdasan Buatan

Kecerdasan buatan berasa dari bahasa Ingggris “*Artificial Intelligence*” atau disingkat AI, yaitu *intelligence* adalah kata sifat yang berarti cerdas sedangkan *artificial* artinya buatan. Kecerdasan buatan yang dimaksud disini merujuk padamesin yang mampu berpikir, menimbang tindakan yang akan diambil, dan

mampu mengambil keputusan seperti yang dilakukan oleh manusia (Azmi and Yasin, 2017).

# Defenisi Kecerdasan Buatan

Menurut (Budiharto and Suhartono, 2014), Kecerdasan buatan atau *Artificial Intelligence* (AI) merupakan bidang ikmu komputer yang mempunyai peran penting di era kini dan masa yang akan datang. Kata *intelligence* berasal dari bahasa latin *intelligo* yang berarti “saya paham”. Jadi, dasar *intelligence* adalah kemampuan memahami dan melakukan aksi.

Menurut Alexander Simon, kecerdasan buatan (Artificial Intelligence) merupakan kawasan penelitian, aplikasi dan intruksi yang terkait dengan pemograman komputer untuk melakukan suatu hal yang dalam pandangan manusia adalah cerdas. Berdasarkan ini, maka kecerdasan buatan menawarkan media maupun uji teori tentang kecerdasan. Teori-teori ini nantinya dapat menyatakan dalam bahasa pemograman dan eksekusinya dapat dibuktikan pada komputer nya (Manalu and Harianja, 2018).

Menurut John MC.Carthy dari Standford, mengusulkan defenisi AI adalah “cabang dari ilmu komputer yang berfokus pada pengembangan computer untuk dapat memilki kemampuan dari perilaku seperti manusia. Dan kecerdasan buatan sebagai kemampuan untuk mencapai sukses dalam menyelesaikan masalah” (Azmi and Yasin, 2017).

# Sejarah Kecerdasan Buatan

*Artificial Intelligence* atau kecerdasan buatan ini merupakan inovasi baru di bidang ilmu pengetahuan. Mulai dari sejak muncul computer modern, yakni pada tahun 1940 dan 1950. Ini kemampuan mesin elektronika baru yang

menyimpan sejumlah besar info, juga memproses dengan kecepatan sangat tinggi menandingi kemampuan manusia (Azmi and Yasin, 2017).

Abad ke-17 sampai abad ke-19

Abad ini merupakan titik awal perkembangan kecerdasan buatan. Hal ini ditandai oleh penemuan-penemuan para ahli tentang kecerdasan buatan.

Blasie Pascal, berhasil menciptakan mesin penghitung digital mekanisme pertama

Charles Babbage dan Ada Lovelace, berhasil membuat mesin perhitungan mekanis yang dapat deprogram.

Bertrand Rusell dan Alfred NorthWhitehead, menerbitkan buku.

Princa Mathematica, yang merombak logika formal.

Welter Pitts, menerbitkan “kalkulus Logis” pada tahun 1943, yang merupakan fondasi untuk jaringan saraf tiruan.

Rene Descartes, mengemukakan bahwa semua tidak ada yang pasti, kecuali kenyataan bahwa seseorang bisa berfikir.

Pada Tahun 1950-1970

Pada 1951, Diertich Prinz berhasil membuat sebuah program permainan catur.

Februari 1951, University of Manchester telah berhasil mengembangkan komputer elektronik pertama di dunia yang diberi nama “Ferranti Mark”.

Tahun 1956, Jhon McCarthy membuat istilah “kecerdasan buatan” pada konferensi pertama yang disediakan untuk pokok persoalan dan menemukan pemrograman Lisp.

Pada tahun 1967, Josep Weinzeibaun berhasil membuat program Eliza. Program ini mampu melakukan terapi terhadap pasien dengan memberikan beberapa pertanyaan.

Marvin Minsky dan Seymour Papert menerbitkan Perceptrons yang merekomendasikan batas aringan saraf tiruan sederhana.

Joel Moses, selama tahun 1960-an dan 1970-an, mendemonstrasikan kekuatan pertimbangan simbolis untuk mengintegrasikan masalah di dalam program Macsyma, program berbasis pengetahuan yang sukses pertama kali dalam bidang matematika.

Pada Tahun 1980-2004

pada tahun 1974, **Paul Jhon Werbos** orang pertama yang menjelaskan jaringan saraf tiruan menggunakan algoritma perambatan balik. Pada 1980-an baru digunakan secara meluas.

**Deep Blue,** Peroleha besar dalam berbagai bidang Ai dan demonstrasi berbagai macam aplikasi berlangsung pada 1990-an.

# Kecerdasan Alami dan Kecerdasan Buatan

Jika dibandingkan dengan kecerdasan buatan alami (kecerdasan yang dimiliki manusia), kecerdasan buatan memiliki keuntungan komersial, antara lain (Sutojo, Mulyanto and Suhartono, 2011).

Kecerdasan buatan lebih bersifat permanen. Kecerdasan alami akan cepat mengalami perubahan. Kemampuan kecerdasan buatan tidak akan pernah berubah selama programnya tidak diubah oleh proggramer. Berbeda dengan kecerdasan alami. Karena sifat manusia yang subjektif, pelupa, dan makin lama makin tua sehingga kemampuan berfikir

berkurang seiring bertambahnya waktu, kemampuan kecerdasan alami cenderung tidak permanen.

Kecerdasan buatan lebih mudah diduplikasi dan disebarkan.

Kecerdasan buatan lebih murah dibandingkan kecerdasan alami.

Kecerdasan buatan bersifat konsisten.

Kecerdasan buatan dapat didokumentasi. Solusi dan keputusan yang dibuat oleh kecerdasan buatan dapat didokumentasi dengan mudah karena disimpan di dalam harddisk dan pencarian datanya relatif lebih mudah dilacak, sedangkan kecerdasan alami hal ini sulit dilakukan.

Kecerdasan buatan dapat mengerjakan pekerjaan lebih cepat dibanding dengan kecedasan alami.

Kecerdasan buatan dapat mengerjakan pekerjaan lebih baik dibanding dengan kecerdasan alami.

Sementara itu, kecerdasan alami memiliki keuntungan sebagai berikut (Sutojo, Mulyanto and Suhartono, 2011):

1. Kreatif, hal ini dikarenakan pengetahuan seorang manusia selalu bertamabah seiring dengan perkembangan waktu.
2. Kecerdasan alami memungkinkan orang menggunakan pengalaman secara langsung. Sedangkan kcerdasan buatan harus bekerja dengan input-input simbolik.
3. Pemikiran manusia dapat digunakan secara luas, sedangkan kecerdasan buatan sangat terbatas.

# Sistem Pakar

Sistem Pakar adalah sistem berbasis komputer yang menggunakan pengetahuan,

dan teknik penalaran dalam memecahkan masalah, yang biasanya dapat diselesaikan oleh seorang pakar dalam bidang tertentu (Fricles Ariwisanto Sianturi, 2019).

# Defenisi Sistem Pakar

Sistem pakar adalah pemograman kecerdasan buatan yang menggabungkan pangkalan pengetahuan base dengan system inferensi untuk meniru seseorang pakar. Sistem pakar merupakan sistem yang berusaha mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh para ahli (Azmi and Yasin, 2017).

Sistem pakar merupakan pengembangan kecerdasan buatan yang menggabungkan pengetahuan dan penelusuran data untuk memecahkan masalah yang secara normal memerlukan keahlian manusia. Tujuan pengembangan sistem pakar sebenanya bukan untuk menggantikan peran manusia, tetapi untuk mensubtitusikan pengetahuan manusia ke dalam bentuk sistem, sehingga dapat digunakan oleh banyak orang (Zm and Erlansari, 2017).

# Konsep Dasar Sistem Pakar

Kecerdasan buatan merupakan cabang ilmu komputer yang bertujuan untuk membuat kinerja komputer dapat berpikir dan bernalar seperti pikiran atau otak manusia. Salah satu cabang dalam ilmu kecerdasan buatan yang banyak dimanfaatkan adalah sistem pakar (Anik Andriani, 2017).

# Kepakaran

Kepakaran merupakan suatu pengetahuan yang diperoleh dari pelatihan, membaca, dan pengalaman. Kepakaran inilah yang memungkinkan para ahli dapat mengambil keputusan lebih cepat dan lebih

baik dari pada seseorang yang bukan pakar. Kepakaran itu sendiri meliputi pengetahuan tentang (Sutojo, Mulyanto and Suhartono, 2011):

* + - * 1. Fakta-fakta tentang bidang permasalahan tertentu.
        2. Teori-teori tentang bidang permasalahan tertentu.
        3. Aturan-aturan dan prosedur-prosedur menurut bidang permasalahan umumnya.
        4. Aturan *heuristic* yang harus dikerjakan dalam suatu situasi tertentu.
        5. Strategi global untuk memecahkan permasalahan.
        6. Pengetahuan tentang pengetahuan (*meta knowledge*).

# Aturan ₋ aturan (*Rule*)

Kebanyakan *software* sistem pakar komersial adalah sistem yang berbasis rule (*rule based system*), yaitu pengetahuan disimpan terutama dalam bentuk *rule,* sebagai prosedur-prosedur masalah (Sutojo, Mulyanto and Suhartono, 2011).

* + - 1. **Pakar (*Expert*)**

Pakar adalah seseorang yang mempunyai pengetahuan, pengalaman, dan metode khusus. Seorang pakar harus mampu menjelaskan dan mempelajari hal – hal baru yang berkaitan dengan topik permasalahan, jika perlu mampu menyusun kembali pengetahuan- pengetahuan yang didapatkan dan dapat memecahkan aturan – aturan serta relevansi kepakarannya. Jadi seorang pakar harus mampu melakukan kegiatan-kegiatan berikut (Sutojo, Mulyanto and Suhartono, 2011):

* + - * 1. Mengenali dan memformulasikan permasalahan.
        2. Memecahkan permasalahan secara cepat dan tepat.
        3. Menerangkan pemecahannya.
        4. Belajar dari pengelaman.
        5. Merestrukturisasi pengetahuan.
        6. Memecahkan aturan-aturan.
        7. Menetukan relevansi.
      1. **Inferensi (*Interferencing*)**

Inferensi adalah sebuah prosedur (program) yang mempunyai kemampuan dalam melakukan penalaran. Inferensi ditampilkan pada suatu komponen yang disebut mesin inferensi yang mencakup prosedur-prosedur mengenai pemecahan masalah. Mesin inferensi berfungsi untuk memandu proses penalaran terhadap suatu kondisi berdasarkan basis pengetahuan untuk mencapai soulsi atau dimilikinya. Mesin inferensi menggunakan strategi pengendalian yang berfungsi sebagai panduan arah dalam melakukan proses penalaran (Sutojo, Mulyanto and Suhartono, 2011).

* + - 1. **Kemampuan Menjelaskan (*Explanation Capability*)**

Fasilitas lain dari sistem pakar adalah kemampuan untuk menjelaskan saran atau rekomendasi yang diberikannya. Penjelasan dilakukan didalam subsistem yang disebut subsistem penjelasan (*explanation*). Bagian dari sistem ini memungkinkan sistem untuk memeriksa penalaran yang dibuatnya sendiri dan menjelaskan operasinya. Ada Perbedaan karakteristik antara sistem pakar dengan sistem Konvensinal. Berikut adalah perbedaan antara sistem pakar dan sistem konvensional pada tabel 2.8 :

# Tabel 2.8 Perbandingan antara Sistem Konvensional dengan Sistem Pakar

|  |  |
| --- | --- |
| **Sistem Konvensional** | **Sistem Pakar** |
| Informasi dan pemrosesannya  biasanya digabungkan dalam satu program | Basis pengetahuan dipisahkan secara jelas dengan mekanisme inferensi. |
| Program tidak membuat kesalahan (yang memebuat kesalahan : pemrograman atau  penggun). | Program dapat berbuat kesalahan. |
| Biasanya tidak menjelaskan mengapa data masukan diperlukan atau bagaimana  output dihasilkan | penjelasan merupakan bagian terpenting dari semua sistem pakar. |
| Perubahan program sangat  menyulitkan. | Perubahan dalam aturan – aturan mudah  untuk dilakukan. |
| Sistem hanya bisa beroperasi setelah lengkap atau selesai. | Sistem dapat beroperasi hanya denagn aturan – aturan yang sedikit (sebagai  prototipe awal). |
| Eksekusi dilakukan langkah demi langkah. | Eksekusi dilakukan denga  menggunakan heuristik dan logika pada seluruh basis pengetahuan. |
| Perlu informasi lengkap agar bisa beroperasi | Dapat beroperasi dengan inforamasi yang tidak lengkap atau mengandung  ketidakpastian. |

|  |  |
| --- | --- |
| Manipulasi efektif dari basis  data yang besar. | Manipulasi efektif dari basis  pengetahuan yang besar. |
| Menggunakan data. | Menggunakan pengetahuan. |
| Tujuan utama : efisiensi | Tujuan utama : efektifitas. |
| Mudah berurusan dengan data  kuantitatif. | Mudah berurusan dengan data kualitatif. |
| Menangkap, menambah, dan  mendistribusikan akses ke data numerik atau informasi. | Menangkap, menambah, dan  mendistribusikan akses ke pertimbangan dan pengetahuan. |

Sumber : Sutojo, Mulyanto and Suhartono, 2011

* + - 1. **Pemindahan Kepakaran (*Transfering Expertise*)**

Tujuan dari sistem pakar adalah memindahkan kepakaran dari seorang pakar kedalam komputer, kemudian ditransferkan kepada orang lain yang bukan pakar, proses ini melibatkan empat kegiatan, yaitu (Sutojo, Mulyanto and Suhartono, 2011):

* + - * 1. Akuisisi pengetahuan (dari pakar atau sumber lain).
        2. Representasi pengetahuan (pada komputer).
        3. Inferensi pengetahuan.
        4. Pemindahan pengetahuan ke pengguna.

# Ciri – ciri dan Area Permasalahan Sistem Pakar

Ciri – ciri dari sistem pakar adalah sebagai berikut (Azmi and Yasin, 2017):

* + - * 1. Terbatas pada domain keahlian.
        2. Memberikan penalaran untuk data yang tidak pasti.
        3. Dapat menjelaskan alas an-alasan dengan cara yang dapat dipahami.
        4. Bekerja berdasarkan kaidah/rule tertentu.
        5. Mudah dimodifikasi.
        6. Basis pengetahuan dan mekanisme inferensi terpisah.
        7. Keluarnya bersifat anjuran.
        8. Sistem dapat mengaktifkan kaidah secara searah yang sesuai.

Menurut (Sutojo, Mulyanto and Suhartono, 2011) sistem pakar menyentuh beberapa area permasalahan sebagai berikut:

1. Interpretasi

Menghasilkan deskripsi situasi berdasarkan data-data masukan.

1. Prediksi

Memperkirakan akibat yang mungkin terjadi dari situasi yang ada.

1. Diagnosis

Menyimpulkan suatu keadaan berdasarkan gejala-gejala yang diberikan.

1. Desain

Melakukan perancangan berdasarkan kendala-kendala yang diberikan.

1. Planning

Merencanakan tindakan-tindakan yang akan dilakukan.

1. Monitoring

Membandingkan hasil pengamatan dengan proses prencanaan.

1. Debugging

Menentukan penyelesaian dari suatu kesalahan sistem.

1. Reparasi

Melaksanakan rencana perbaikan.

1. Instruction

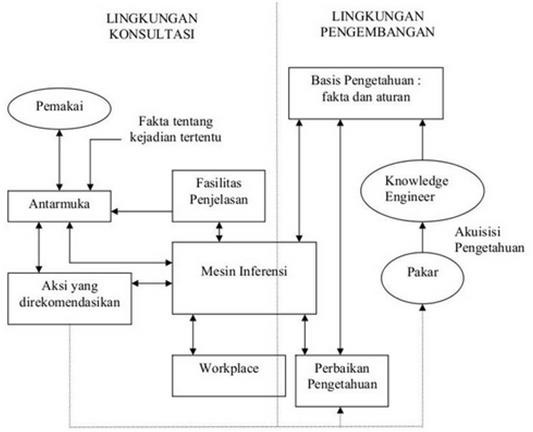
Melakukan instruksi untuk diagnosis, debugging, dan perbaikan kerja.

1. Kontrol

Melakukan kontrol terhadap hasil interpretasi, diagnosis, debugging, monitoring, dan perbaikan tingkah laku sistem.

# Struktur Sistem Pakar

Ada dua bagian penting dari sistem pakar, yaitu lingkungan pengembang (*development environtment*) dan lingkungan konsultasi (*consultation wnvirontment*). Lingkungan pengembang digunakan oleh pembuat sistem pakar untuk membangun komponen-komponennya dan memperkenalkan pengetahuan ke dalam *knowledge base* (basis pengetahuan). Berikut adalah gambar yang menunjukkan komponen-komponen yang penting dalam sebuah sistem pakar.



Sumber : Azmi and Yasin, 2017

# Gambar 2.8 Komponen - komponen yang penting dalam sistem pakar

Keterangan :

Akuisi pengetahuan

Subsistem ini digunakan untuk memasukan pengetahuan dari seorang pakar dengan cara merekayasa pengetahuan agar bisa diproses oleh komputer dan menaruhnya ke dalam basis pengetahuan dengan format tertentu (dalam bentuk representasi pengetahuan). Sumber-sumber pengetahuan bisa diperoleh dari pakar, buku, dokumen, multimedia, basis data, laporan riset, dan informasi yang terdapat di web.

Basis pengetahuan (*Knowledge Base*)

Basis pengetahuan mengandung pengetahuan yang diperlukan untuk memahami, memformulasikan dan menyelesaikan masalah. Basis pengetahuan terdiri dari dua elemen dasar, yaitu:

Fakta, misalnya situasi, kondisi, atau permasalahan yang ada.

*Rule* (aturan), untuk mengarahkan pengguna pengetahuan dalam memecahkan masalah.

Mesin Inferensi (*Inference Engine*)

Mesin inferensi merupakan sebuah program yang berfungsi untuk memandu dan proses penalaran terhadap suatu kondisi berdasarkan pada basis pengetahuan yang ada, memanipulasi dan mengarahkan kaidah, model, dan model fakta yang disimpan dalam basis pengetahuan untuk mencapai solusi atau kesimpulan. Dalam prosesnya, mesin inferensi menggunakan strategi pengendalian, yaitu strategi yang berfungsi sebagai panduan arah dalam melakukan proses penalaran.

Dearah Kerja (*Blackboard*)

Untuk merekam hasil sementara yang akan dijadikan sebagai keputusan dan untuk menjelaskan sebuah masalah yang sedang terjadi, sistem pakar membutuhkan *Blackboard,* yaitu area pada memori yang berfungsi sebagai basis data. Terdapat tiga tipe keputusan yang dapat direkam pada *blackboard,* yaitu:

Rencana : bagaimana menghadapi masalah.

Agenda :aksi-aksi potensial yang sedang menunggu untuk dieksekusi.

Solusi : calon aksi yang akan dibangkitkan.

Antarmuka Pemakai (*User Inference*)

*User inference* digunakan sebagai media komunikasi antara pengguna dan sistem komunikasi disajikan dalam bahasa alami (*netural language*) dan dilengkapidengan grafik, menu, dan formulir

elektronik. Pada bagian ini akan terjadi dialog antara sistem pakar dan pengguna.

Sub sistem penjelasan (*Explanation Subsystem / Justifier*)

Berfungsi memberikan penjelasan kepada peengguna, kesimpulan dapat diambil. Kemampuan seperti ini sangat penting bagi pengguna untuk mengetahui proses pemindahan keahlian pakar maupun dalam pemecahan masalah.

Sistem Perbaikan Pengetahuan (*Knowledge Refining System*) Kemampuan memperbaiki pengetahuan (*knowledge refining system*) dari seorang pakar diperlukan untuk menganalisis pengetahuan, belajar dari kesalahan masa lalu, kemudian memperbaiki pengetahuannya sehinggadapat dipakai pada masa mendatang.

Pengguna (*User*)

Pada umumnya pengguna sistem pakar bukanlah seorang pakar (*non- expert*) yang membutuhkan solusi, saran, atau pelatihan (*training*) dari berbagai permasalahanya yang ada.

# Kelebihan dan Kekurangan Sistem Pikir

Beberapa kelebihan dari sistem pakar adalah sebagai berikut (Azmi and Yasin, 2017):

Menghimpun data dalam jumlah besar.

Menyimpan data tersebut dalam jangka waktu yang lama dalam bentuk tertentu.

Mengerjakan perhitungan secara cepat dan tepat serta mencari kembali data yang tersimpan dengan kecepatan tinggi.

Meningkatkan produktifitas.

Membuat seseorang yang awam bekerja seperti layaknya seorang pakar.

Meningkatkan kualitas, dengan memberi nasehat yang konsisten dan mengurangi keselahan.

Mampu menangkap pengetahuan dan kepakaran seseorang.

Dapat beroperasi dilingkungan yang berbahaya.

Memudahkan akses pengetahuan seorang pakar.

Andal.

Meningkatkan kapabilitas sistem komputer.

Mampu bekerja dengan informasi yang tidak lengkap atau tidak pasti, selama konsultasi dengan sistem pakar tetap akan memberikan jawabannya.

Bisa digunakan sebagai media pelengkap dalam pelatihan. Pengguna pemula yang bekerja dengan sistem pakar akan menjadi lebih berpengalaman karena adanya fasilitas penjelasan yang berfungsi sebagai guru.

Meningkatkan kemampuan untuk menyelesaikan masalah – masalah yang terjadi, karena mengambil dari sumber pengetahuan banyak pakar.

Disamping memiliki kelebihan, sistem pakar juga mempunyai beberapa kekurangan sebagai berikut (Sutojo, Mulyanto and Suhartono, 2011):

1. Biaya yang sangat mahal untuk membuat dan memeliharanya.
2. Sulit dikembangkan karena keterbatasan keahlian dan ketersediaan pakar
3. Sistem pakar tidak 100% bernilai benar.

# Kelebihan dan Kekurangan Sistem Pikir

Beberapa keuntungan dari sistem pakar adalah sebagai berikut (Azmi and Yasin, 2017):

Menjadikan pengetahuan lebih mudah didapat.

Meningkatkan *output* dan produktifitas.

Menyimpan kemampuan dan keahlian para pakar.

Meningkatkan penyelesaian permasalahan.

Meningkatkan reabilitas.

Memberikan *respons* (jawaban) yang cepat.

Merupakan panduan yang *intelligence* (cerdas).

Dapat bekerja dengan informasi yang kurang lengkap dan mengandung ketidakpastian.

Dapat digunakan untuk mengakses basis data dengan cara cerdas.

## Naïve Bayes

*Naïve Bayes* merupakan sebuah metoda klasifikasi menggunakan metode probabilitas dan statistik. metode bayes memanfaakan data sampel yang diperoleh dari populasi juga memperhitungkan suatu distribusi awal yang disebut distribusi prior

**Defenisi *Naïve Bayes***

*Naive Bayes* merupakan salah satu metode peluang sederhana yang berdasarkan pada penerapan *Teorema Bayes* dengan asumsi antar variabel penjelas saling bebas (independen). Algoritma ini memanfaatkan metode probabilitas dan statistic yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi probabilitas dimasa depan berdasarkan pengalaman di masa sebelumnya (Zm and Erlansari, 2017).

Penerapan pada *teorema Bayes* memungkinkan seseorang untuk mempengaruhi keyakinannya mengenai parameter setelah data diperoleh. Sehingga dalam hal ini mengharuskan adanya keyakinan awal (*prior*) sebelum memulai inferensi (Azmi and Yasin, 2017).

**Rumus yang digunakan pada *Naïve Bayes***

Probabilitas bayes merupakan salah satu cara untuk mengatasi ketidakpastian data dengan menggunakan formula bayes yang dinyatakan:

P(H|E) =

P(E|H) x P(H) P(E)

..........................................(1)

Keterangan:

P(H|E) :Probabilitas hipotesa H terjadi jika evidence E terjadi. P(E|H) :Probabilitas munculnya evidence E, jika hipotesaH terjadi. P(H) :Probabilitas hipotesa H tanpa memandang evidence apapun. P(E) :Probabilitas evindece E tanpa memandang apapun.

Atau juga bisa menggunakan rumus berikutnya jika terdapat Evidence Tuggal E dan Hipotesa Ganda H :

P(Hi|E) =

P(E|Hi) x (Hi)

∑n P(E|Hi) x P(Hi)

k=1

..........................................(2)

Keterangan :

P(Hi|E) :Probabilitas hipotesa Hi terjadi jika evidence E terjadi.

P(E|Hi) :Probabilitas munculnya evidence E jika diketahui hipotesa Hi terjadi. P(Hi) :Probabilitas hipotesa Hi, tanpa memandang evidence apapun.

n :Jumlah hipotesa yang terjadi.

Ada juga rumus Teorema Bayes jika terdapat Evidence Ganda E dan Hipotesa Ganda H :

P(Hi|E1E2 … En) =

P(E1|Hi) x P(E2|Hi) x … x P(Em|Hi) x P(Hi)

∑n P(E1|Hk) x P(E2|Hk)x … x P(Em|Hk) x P(Hk)

k=1

..........................................(3)

# Website

Menurut Yuhefizar (dalam Prayitno and Safitri, 2015), *website* adalah keseluruhan halaman-halaman web yang terdapat dari sebuah domain yang mengandung informasi.

**Defenisi *Website***

Menurut Agus Hariyanto (dalam Destiningrum and Adrian, 2017), *Website* adalah “Web dapat diartikan sebagai kumpulan halaman yang menampilkan informasi data teks, data gambar, data animasi, suara, video dan gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (hyperlink)”.

Halaman-halaman website tersebut diakses oleh pengguna melalui protokol komunikasi jaringan yang disebut sebagai *HypertextTransferProtocol* (HTTP). Untuk meningkatkan aspek keamanan dan aspek privasi ynag lebih baik, situs web dapat pula mengimlementasikan mekanisme pengaksesan

melalui protocol HTTPS (*HypertextTransferProtocolSecure*) (Mandala, Eka Praja Wiyata, 2015).

**Jenis - Jenis *Website***

Secara umum, website digolongkan tiga jenis, yaitu (Mandala, Eka Praja Wiyata, 2015):

*Website* Statis

*Website* statis adalah *website* yang memiliki halaman yang tidak berubah. Artinya adalah untuk melakukan perubahan pada suatu halaman dilakukan secara manual dengan mengubah kode yang menjadi struktur dari situs itu.

*Website* Dinamis

*Website* Dinamis merupakan *website* yang secara spesifik didesain agar isi yang terdapat dalam situs tersebut dapat diperbaharui secara berkala dengan mudah. Sesuai dengan namanya, isi yang terkandung dalam website ini umumnya berubah setelah melewati suatu periode tertentu. Website ini secara struktur diperuntukan untuk bisa melakukan perubahan sesering mungkin. Biasanya selain halaman utama yang bisa diakses oleh user pada umumnya, juga disediakan halaman *backend* untuk mengedit konten dari website. Misalnya web berita atau web portal.

*Website* Interaktif

*Website* interaktif adalah website yang saat ini memang sedang banyak digemari. User bisa berinteraksi dan beradu argumen mengenai apa yang menjadi pemikiran mereka. Biasanya website seperti memiliki moderator untuk mengatur supaya topik yang

diperbincangkan tidak melenceng dari alur pembicaraan. Salah satu contoh website interaktif adalah blog dan forum.

## Webserver dan Webbrowser

*Web server* itu dapat dikatakan sebagai suatu program komputer yang memiliki tugas menerima permintaan HTTP dari komputer klien, yang dikenal dengan nama *web browser* dan melayani mereka dengan menyediakan respon HTTP berupa konten data, biasanya berupa halaman web yang terdiri dari dokumen HTML dan objek yang terkait seperti gambar, text, suara, dan sebagainya (Mandala, Eka Praja Wiyata, 2015).

# Pemrograman Web

Pada tahap pembuatan website, pemrograman web ini menggunakan bahasa pemrograman PHP dan MySQL.

**PHP (*Hypertext Preprocessor*)**

PHP atau kependekan dari *Hypertext Preprocessor* adalah salah satu bahasa pemrograman open source yang sangat cocok atau dikhususkan untuk pengembangan web dan dapat ditanamkan pada sebuah skripsi HTML. Bahasa PHP dapat dikatakan menggambarkan beberapa bahasa pemrograman seperti C, Java, dan Perl serta mudah untuk dipelajari. PHP merupakan bahasa scripting server – side, dimana pemrosesan datanya dilakukan pada sisi server. Sederhananya, serverlah yang akan menerjemahkan skrip program, baru kemudian hasilnya akan dikirim kepada client yang melakukan permintaan (Firman *et al.*, 2016).

*Hypertext Preprocessor* (PHP) adalah bahasa skrip yang dapat ditanamkan disisipkan ke dalam HTML. PHP banyak dipakai untuk membuat situs web

dinamis. PHP dapat digunakan untuk membangun sebuah CMS (*Content Management System*) (Mandala, Eka Praja Wiyata, 2015).

# Keunggulan PHP

Menurut (Mandala, Eka Praja Wiyata, 2015) ada beberapa keunggulan PHP sebagai berikut:

PHP berjalan pada berbagai *platform* (Windows, Linux, Unix, Mac OS, dll).

PHP kompatibel dengan hampir semua *server* yang digunakan sekarang ini (Apache, IIS, dll).

PHP mendukung berbagai *database*.

PHP itu gratis. *Download* dari sumber resmi PHP:[www.php.net](http://www.php.net/)

PHP mudah dipelajari dan berjalan efisien pada isi *server*.

# Database

Menurut Fathansyah (dalam Rozaq, Lestari and Handayani, 2015) , Basis Data (Database) terdiri atas 2 kata, yaitu basis dan data. Basis kurang lebih dapat diartikan sebagai markas atau gudang, tempat bersarang/berkumpul. Sedangkan Data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang diwujudkan dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya.

Database *Server* adalah sebuah program komputer yang menyediakan layanan pengolahan basis data dan melayani komputer atau program aplikasi basis data yang menggunakan model kline/server*.* Istilah ini juga merujuk kepada sebuah komputer (umumnya merupakan server) yang didedikasikan

untuk menjalankan program yang bersangkutan. *Database Management System* (DBMS) pada umumnya menyediakan fungsi – fungsi server basis data, dan beberapa DBMS (seperti halnya MySQL atau Microsoft SQL Server) sangat bergantung kepada model klien - server untuk mengakses basis datanya (Mandala, Eka Praja Wiyata, 2015).

Menurut (Handayani *et al.*, 2018) Beberapa keunggulan dari penggunaan DBMS yaitu:

Mengendalikan atau mengurangi duplikasi data.

Menjaga konsentrasi dan integrasi data.

Memudahkan perolehan informasi yang lebih banyak.

# MySQL

MySQL adalah program aplikasi database yang berbasis *open source*. MySQL mampu menangani database yang kompleks dan cukup besar. MySQL juga dapat menangani database *client server*. MySQL AB membuat MySQL sebagai perangkat lunak gratis di bawah lisensi GNU *General Public License* (GPL), tetapi mereka juga menjual di bawah lisensi komersial untuk kasus – kasus dimana penggunaannya tidak cocok dengan penggunaan GPL (Mandala, Eka Praja Wiyata, 2015).

Beberapa keistimewaan yang dimiliki MySQL sebagi berikut (Mandala, Eka Praja Wiyata, 2015):

Portabilitas, MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.

Open Source Software, MySQL didistribusikan sebagai perangkat lunak sumber terbuka, dibawah lisensi GPL sehingga dapat digunakan secara gratis

Multi-User, MySQL dapat digunakan oleh beberapa pengguna dengan waktu yang bersamaan tanpa mengalami masalah atau konflik.

Performence Tuning, MySQL memiliki kecepatan yang menakjubkandalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.

Ragam tipe data, MySQL memiliki ragam tipe data yang sangat kaya, seperti signed/unsigned integer, float, double, char, text, date, timestamp, dan lain-lain.

Perintah dan fungsi, MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah select dan where dalam perintah.

Kemanaan, MySQL memiliki beberapa lapisan keamanan seperti level subnetmask, nama host, dan izin akses user dengan system akses perizinan yang menditeil serta sandi terenskripsi.

Skalabilitas dan pembatasan, MySQL mampu menangani basis data dalam skala besar, dengan jumlah rekaman (*records*) lebih dari 50 juta data dan 60 ribu tabel serta 5 miliar baris.

Konektivitas, MySQL dapat melkukan konektivitas dengan klien menggunakan protocol TCP/IP, *Unix Soket* (UNIX), atau *NamedTipes* (NT).

Lokalisasi, MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meskipun bahasa indonesia belum termasuk didalamnya.

Antarmuka, MySQL memiliki antar muka (*Interface*) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*)

Klien dan peralatan, MySQL dilengkapi dengan berbagai peralatan (*tool*) yang dapat digunakan untuk administrasi basis data, dan pada setiap peralatan yang ada disertakan petunjuk *online*.

Struktur tabel, MySQL memiliki struktur tabel yang lebih fleksibeldalam menangani ALTER TABLE, dibandingkan basis data lainnya semacam PostdreSQL ataupun Oracle.

# Gonore

Gonore merupakan jenis penyakit menular seksual (PMS) klasik yang disebabkan oleh infeksi bakteri *Neisseria gonorrhoeae*.

# Defenisi Gonore

Gonore adalah penyakit menular seksual yang disebabkan oleh bakteri *Neisseria gonorrhoeae* yang menginfeksi lapisan dalam saluran kantung kemih, leher rahim, rektum, tenggorokan, serta bagian putih mata. Penyakit ini menyebar melalui aliran darah ke bagian tubuh lainnya terutama pada kulit dan persendian dan pada umumnya penularannya melalui hubungan seksual. Penyakit gonore ini tidak hanya menyerang pria dan wanita dewasa. Namun bayi yang baru lahir sekalipun bisa terinfeksi gonore dari ibunya bila selama proses kelahiran terjadi pembengkakan pada kedua kelopak matanya dan mengeluarkan nanah. Jika tidak di obati maka akan menyebabkan kebutaan pada bayi (Susanto, Ari and GA, 2013).

# Jenis – Jenis Penyakit Gonore

Beberapa penyakit yang ditimbulkan oleh Gonore:

Uretritis Gonore

Menururt CDC (dalam Dwi Harningtyas, 2017) Uretritis pada pria adalah suatu peradangan yang ditandai dengan keluarnya cairan bernanah pada uretra kadang disertai nyeri saat buang air kecil. Sedangkan pada wanita gejala umumnya ialah nyeri saat buang air kecil dan pada pemeriksaan uretra eksternum tampak merah. Menurut Daili dan Nilasari (dalam Dwi Harningtyas, 2017) penulran terjadi melalui kontak seksual dengan penderita gonore. Masa inkubasi penyakit pada pria sangat singkat dan pada umumnya bervariasi antara 2-8 hari, denagn kebanyakan infeksi menjadi simptomatik dalam 2 minggu. Masa inkubasi pada wanita sulit ditentukan karena pada umunya asimptomatik, dan baru diketahui penderita setelah terjadi komplikasi.

Servisitis Gonore

Servisitis pada wanita memiliki banyak fitur yang sama dengan uretritis pada pria dan banyak kasus disebabkan oleh IMS (Infeksi Menular Seksual). Pada gangguanya ini mempengaruhi sekitar 60% perempuan karena infeksi bakteri *Neisseria gonorrhoeae* (Abrori, Hermawan and Inayati, 2016). Menurut Duenhoelter (dalam Abrori, Hermawan and Inayati, 2016) Servisitis adalah suatu proses peradangan yang melibatkan *epitel serviks*. Ketika terjadina radang dari selaput lendir saluran *servikal*. Singkatnya, servisitis adalah peradangan dari *serviks uterus*.

Faringitis Gonore

Cara infeksi melalui orofaring atau disebut juga dengan bagian belakang faring yang terletak dibelakang rongga mulut. Faringitis gonore menyebar karena adanya kontak hubungan seksual melaui mulut (*Oreal Sex*) dengan seseorang penderita gonore pada tenggorokan (Susanto, Ari and GA, 2013).

Proktitis Gonore

Wainta dan pria homoseksual yang melakukan hubungan seksual melalui anus (lubang dubur) bisa menderita gonore pada rektumnya. Penderita merasakan tidak nyaman di sekitar anusnya dan keluarnya cairan dari rektum. Daerah sekitar anus tampak merah dan kasar, tinjanya terbungkus oleh lendir dan nanah (Susanto, Ari and GA, 2013).

# Jenis dan gejala yang ditimbulkan penyakit Gonore

Uretritis Gonore

Gejala Uretritis Gonore yang timbul pada pria (Daili, 2010) :

Panas dibagian uretra

Rasa terbakar ketika buang air kecil

Kencing berdarah

Lubang penis bengkak dan merah

Mengeluarkan cairan atau nanah

Kelenjer getah bening bengkak diabgian selangkangan

Nyeri ketika berhubungan seksual

Nyeri perut

Demam dan mengigil

Nyeri panggul

Rasa terbakar ketika buang air kecil

Nyeri ketika berhubungan seksual

Keluarnya cairan keputihan dari vagina

Servisitis Gonore

Berikut adalah gejala pada servisitis gonore (Susanto, Ari and GA, 2013):

Vagina mengeluarkan cairan berwarna kuning pucat keabu- abuan

Vagina berbau tidak sedap

Sering buang air kecil dan menyakitkan

Nyeri saat berhubungan intim

Pendarahan dari vagina setelah berhubungan seksual

Vagina terasa nyeri

Panggul terasa nyeri

Sakit punggung

Nyeri dibagian panggul atau perut

Demam

Bengkak pada area lubang kencing

Faringitis Gonore

Berikut adalah gejala pada faringitis gonore (Susanto, Ari and GA, 2013):

Nyeri tenggorokan

Nyeri menelan

Tenggorokan kemerahan dengan bercak putih

Timbulnya benjolan dileher

Sakit tenggorokan yang datang terus menerus

Batuk

Demam

Tenggorokan terdapat cairan berwarna keputihan kekuningan

Proktitis Gonore

Berikut adalah gejala pada proktitis gonore (Susanto, Ari and GA, 2013):

Anus Gatal

Jarang Buang air besar

Rasa sakit saat BAB

Keluar cairan asing dari anus

Benjolan nanah di dalam anus

# BAB III METODOLOGI PENELITIAN

# Kerangka Penelitian

Kerangka penelitian ini merupakan urutan yang dilakukan dalam penelitian. Agar langkah-langkah yang diambil penulis dalam perencangan ini tidak melenceng dari pokok pembahasan. Adapun kerangka penelitian yang penulis lakukan dalam penelitian yang akan diuraikan pada Gambar 3.1 sebagai berikut :

Penelitian Pendahuluan

Analisa

P. Labor

P. Pustaka

P. Lapangan

Wawancara

Pengumpulan Data

Perancangan

Implementasi

Deployment

Activity

Statechart

Collaboration

Sequence

Class

Use Case

Unified Modeling Language

Pengujian

# Gambar 3.1 Kerangka Penelitian

# Tahapan Penelitian

Pada tahapan penelitian ini berisi tentang penjelasan kerangka penelitian diatas.

Dimana pada kerangka penelitian tersebut berisikan tentang penelitian pendahuluan, pengumpulan data, analisa, perancangan, implementasi, dan pengujian. Dalam melakukan penelitian, diperlukan metode untuk melakukan penelitian tersebut. Metode penelitian adalah serangkaian cara / kegiatan pelaksanaan penelitian menurut langkah-langkah yang telah disusun secara sistematis dan logis sehingga dapat dijadikan pedoman yang jelas dan mudah untuk menyelesaikan suatu permasalahan. Tahapan dari kerangka penelitian termasuk bagian yang saling berkaitan untuk menentukan proses berikutnya.

# Penelitian Pendahuluan

Pada tahapan ini, penelitian pendahuluan adalah langkah pertama dalam melakukan penelitian dengan cara menganalisa terlebih dahulu masalah-masalah yang akan diteliti. Penelitian ini dilakukan dengan melakukan wawancara. Penelitian ini bertujuan untuk membuat sistem pakar berbasis web yang dapat digunakan untuk mendiagnosa penyakit gonore dengan menggunkanan metode *naïve bayes*. Dan dapat memudahkan masyarakat untuk menemukan solusi dan cara mengatasi penyakit gonore lebih awal sehingga tidak terlambat untuk diobati.

# Pengumpulan Data

Pada tahapan kedua merupakan tahap pengumpulan data yang dilakukan setelah penelitian pendahuluan yang berisi urutan-urutan mulai dari awal sampai akhir penelitian. Dalam melakukan penelitian, untuk mendapatkan informasi seputar objek penelitian diperlukan tahap pengumpulan data sebagai berikut:

# Waktu Penelitian

# Tabel 3.1 Jadwal Penelitian

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Kegiatan** | **Bulan** | | | | | | | | | | | | | | | | | | | |
| **September**  **-**  **Desember** | | | | **Februari** | | | | **Maret** | | | | **April** | | | | **Mei** | | | |
| **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** |
| 1 | Perancangan Penelitian |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 | Pengumpulan  Data |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 | Analisa data dan sistem |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Perancangan Aplikasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 | Pembuatan Aplikasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 | Uji Coba |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7 | Implementasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 8 | Bimbingan  Skripsi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 9 | Penyusunan Skripsi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

* + - 1. **Tempat Penelitian**

Tempat untuk melakukan penelitian dan pengambilan data pada objek penelitian adalah Rumah Sakit Ibnu Sina Padang dengan pakar spesialis kulit dan kelamin yang bernama Dr. Qaira Anum, Sp,KK di Rumah Sakit Ibnu Sina Padang.

# Metode Penelitian

Untuk menyelesaikan permasalahan yang ada metode pengumpulan data yang dilakukan penulis sebagai berikut:

* + - * 1. Penelitian Lapangan (*Field Reaserch*)

Penelitian ini dilakukan dilapangan yaitu mempelajari metode- metode yang bersangkutan dengan masalah yang dihadapi.

Pengamatan (*Observation*)

Pada tahap ini dilakukan pengamatan lapangan dengan cara mempelajari permasalahan tentang diagnosa penyakit Gonore yang diderita dan dirasakan.

Wawancara (*Interview*)

Pada tahap ini dilakukan wawancara atau interview dengan seorang pakar yang bernama Dr. Qaira Anum, Sp,KK. Wawancara dilakukan di Rumah Sakit Ibnu Sina Padang. Untuk mendapaktan informasi tentang gejala penyakit Gonore.

* + - * 1. Penelitian Pustaka (*Library Reaserch*)

Pada tahap ini merupakan tindak lanjutan dari penelitian lapangan dengan cara mempelajari buku-buku dan jurnal yang berhubungan dengan masalah yang dhiadapi. Hal ini dimaksudkan agar penulis mendapatkan data-data yang berhubungan dengan objek penelitian

berdasarkan buku-buku, jurnal, internet sehingga mendapatkan informasi yang lebih akurat.

* + - * 1. Penelitian Laboratorium (*Laboratorium Reaserch*)

Pada tahap penelitian ini merupakan suatu metode penelitian yang dilakukan dengan menggunakan alat bantu personal komputer (PC). Alat bantu tersebut digunakan untuk merancang program sistem pakar yang sesuai dengan topik permasalahan yang diteliti. Digunakan untuk mempraktekan langsung hasil analisa untuk menguji keakuratan sistem yang akan digunakan.

Adapun spesifikasi *hardware* dan *software* yang digunakan dalam melakukan pembuatan perangkat lunak ini adalah sebagai berikut:

1. Perangkat Keras (*Hardware*)
   1. Laptop Asus X441U
   2. *Processor* Intel Core i3-6006U, 2.0GHz
   3. *Memory* 4 GB
   4. *Harddisk* 1 TB
   5. *Flashdisk* kapasitas 32 GB
2. Perangkat Lunak (*Software*)
   1. *Operating System : Windows 10 Home Single Language*
   2. *Microsoft Office* 2010
   3. *Mowes Portable*
   4. *MySQL Server*
   5. *Macromedia Dreamweaver* 8
   6. *Astah Community*
   7. *Chrome*

# Analisa

Dalam tahap ini dapat dilakukan dengan tiga tahap analisa sebagai berikut:

1. Analisa Data

Data-data yang didapat dari hasil wawancara dengan seorang pakar atau spesialis kemudian akan dilakukan analisis sehingga diperoleh berupa hasil menentukan gejala, jenis, dan cara pengobatannya dari penyakit Gonore.

1. Analisa Proses

Naïve Bayes adalah metode yang akan digunakan dalam penelitian untuk menghasilkan solusi atau masalah, dimana metode *Naïve Bayes* memprediksi peluang dimasa depan berdasarkan pengalaman sebelumnya sehingga dapat kita kenal sebagai teorema bayes.

1. Analisa Sistem

Pada tahap analisa sistem dilakukan untuk merancang atau membangun sistem yang akan dibuat dengan menggunakan metode *Naïve Bayes*, perancangan sistem pakar tersebut meliputi merancang sebuah tampilan *user*, merancang basis data untuk sistem agar manajemen file lebih teratur, merancang *coding* program, dan program tersbut akan dirancang menggunakan bahasa pemrograman PHP dan MySQL, dan barulah terbentuk suatu sistem pakar yang bisa digunakan.

# Perancangan

Pada perancangan ini ada dua jenis tahapan perancangan yang akan dilakukan diantaranya adalah sebagai berikut:

1. Peracangan Model

Pada tahapan ini digunakan UML (*Unified Modeling Language*) adalah

suatu model logika data yang dibuat untuk menggambarkan dari mana asal data, kemana tujuan data yang akan keluar dari sistem, dimana data tersebut dapat disimpan, dan proses apa yang menghasilkan data tersebut, interaksinya antara data yang tersimpan dan proses yang dikenakan pada data tersebut. Berikut adalah model UML (*Unified Modeling Language*) yang digunakan:

* 1. *Use Case Diagram* menggambarkan fungsionalitas yang diharapkan dari sebuah sistem pakar untuk merepresentasikan sebuah interaksi antara aktor dengan sistem pakar tersebut. Misalnya aktor melakukan konsultasi penyakit Gonore yang dialami dengan cara berinteraksi dengan sistem pakar tersebut.
  2. *Class Diagram* Mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat diantara sistem, seperti *user* dengan admin, kelas *user* dengan penyakit, kelas admin dengan solusi.
  3. *Sequence Diagram* merupakan salah satu diagram yang memperlihatkan pengorganisasian interaksi yang terdapat disekitar objek dan hubungannya terhadap yang lainnya. Proses berjalannya operasi diurutkan dari kiri kekanan berdasarkan waktu terjadinya dalam pesan yang terurut. *Sequence diagram* bisa digunakan untuk menggambarkan skenario dan rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah *event* untuk menghasilkan *output* tertentu. Misalnya admin melakukan penghapusan data *user*, admin melakukan penambahan, pengeditan, penghapusan data gejala penyakit, admin melakukan penambahan data gejala penyakit

dan melakukan penambahan solusi.

* 1. *Collaboration* menggambarkan interaksi antara objek seperti *sequence diagram,* tetapi lebih menekankan pada peran masing- masing objek dan buka pada waktu penyampaian *message.* Misalnya *collaborationuser*, *collaboration* gejala, *collaboration* solusi, *collaboration* pertanyaan dan *collaboration* konsultasi.
  2. *Statechart Diagram* menggambarkan transisi dan perubahan keadaan dari sistem pakar dan objeknya. Seperti bagaimana proses penambahan data *user*, proses penambahan data penyakit Gonore, proses penambahan data pertanyaan dan solusi dari admin.
  3. *Activity Diagram* adalah teknik untuk menggambarkan logika procedural, dan jalur kerja. Diagram ini berperan mirip dengan aliran aktifitas dalam sistem pakar yang dirancang. *Activity diagram user* menggambarkan segala aktivitas yang dilakukan oleh *user* terhadap sistem pakar dan *activity diagram* admin menggambarkan segala aktivitas yang biasa dilakukan admin terhadap sistem pakar.
  4. *Deployment Diagram* merupakan gambaran proses-proses berbeda pada suatu sistem yang berjalan dan bagaimana relasi didalamnya. Dan digunakan untuk menggambarkan detail bagaimana komponen disusun di infrastruktur sistem pakar.

1. Perancangan *Interface*

Perancangan *interface* atau perancangan antar muka merupakan bentuk dari rancangan tampilan sementara dari pembuatan aplikasi sistem pakar. Perancangan ini dibuat agar dapat memberikan keterangan atau penjelasan tentang tampilan yang dihadapi oleh aktor pada saat

menggunakan sistem, sehingga dapat mempermudah dalam mengimplementasikan aplikasi dan mempermudah dalam pembangunan aplikasi.

# Implementasi

Implementasi sistem merupakan bagian dari siklus hidup pengembangan sistem. pada tahapan ini akan dirancang aplikasi sistem pakar khusus penyakit Gonore menggunakan bahasa pemrograman PHP dan MySQL dengan menerapakan metode *Naïve Bayes*.

# Pengujian

Pengujian sistem merupakan tahap melakukan *testing* untuk mengetahui kesalahan-kesalahan dan memastikan sistem sistem pakar dapat memberikan hasil yang akurat. Tujuan dilakukan pengujian ini untuk memastikan apakah sistem yang dibuat dapat bekerja sesuai dengan apa yang telah direncanakan sebelumnya. Pengujian sistem dilakukan dengan cara sebagai berikut:

1. LAN (*Local Area Network*)

Peneliti hanya akan memonitoring melalui PC dengan tampilan *web* yang telah dirancang melalui akses *Locak Area Network* (LAN), dimana komputer *server* dan komputer *client* terhubung satu sama lain dengan menggunakan kabel jaringan dan memberikan IP *Address* ke masing-masing komputer tersebut.

1. Online

Pembuatan program ini akan bisa diakses secara online dengan dihosting terlebih dahulu yang mendukung format PHP dan MySQL seperti *hostinger.com.* selanjutnya membuat sebuah domain sesuai

dengan nama yang diinginkan agar *web* dan *database* bisa di *upload* ke *web hosting* tersebut. Setelah semua proses dilakukan maka *website* akan bisa diakses oleh pengguna yang ingin memakai aplikasi ini dengan cara langsung membuka alamat *web* yang telah di *hosting.*

1. Interface

Tahapan model interface dilakukan dengan pengamatan secara langsung oleh pengguna interaksi secara langsung dengan model interface yang disajikan dalam bentuk *prototype.* Proses ini dilakukan agar kesalahan dapat terdekteksi dan di rubah secara cepat.

1. Aplikasi

Aplikasi dapat dinyatakan sebagai proses validasi dan verifikasi bahwa aplikasi yang dirancang dapat memenuhi kebutuhan yang mendasari perancangan aplikasi tersebut dan berjalan sesuai dengan diharapkan.

# BAB IV

**BIAYA DAN JADWAL PENELITIAN**

# Anggaran Biaya

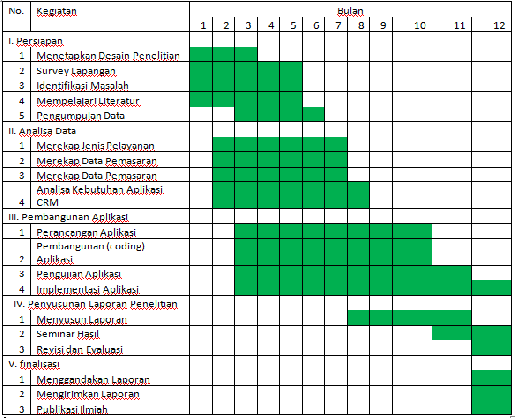
Penelitian ini dilakukan dalam jangka waktu selama 1 (satu) tahun dengan rincian kegiatan sebagaimna dijelaskan melalui matriks table berikut ini.

# Tabel 1. Rencana Anggaran Biaya Penelitian

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Jenis Pengeluaran | Biaya yang diusulkan | | |
| Tahun 1 | Tahun  2 | Tahun  3 |
| 1 | Gaji Ketua dan  Anggota | 2.955.000,- |  |  |
| 2 | Bahan Habis Pakai | 2.970.000,- |  |  |
| 3 | Luaran Penelitian | 4.925.000,- |  |  |
|  | **TOTAL** | **10.850.000,-** |  |  |

* 1. **Jadwal Penelitian**

# Tabel 2. Jadwal Penelitian



# DAFTAR PUSTAKA

Azmi, Z. and Yasin, V. (2017) „Pengantar Sistem Pakar dan Metode‟, *Jakarta: Mitra Wacana Media*.

Budiharto, W. and Suhartono, D. (2014) „Artificial Intelligence Konsep dan Penerapannya‟, *Yogyakarta: Andi*.

Daili, S. F. (2010) „Infeksi Menular Seksual dalam Ilmu Penyakit Kulit dan Kelamin, Edisi ke-6‟. Penerbit FKUI, Jakarta.

Anik Andriani, M. (2017) „Kom, Pemrograman Sistem Pakar Konsep Dasar dan Aplikasinya Menggunakan Visual Basic 6‟, *Yogyakarta: MediaKom*.

Handayani, P. W. *et al.* (2018) „Pengantar sistem informasi manajemen rumah sakit (SIMRS)‟. Rajawali Pers.

Janner, S. (2010) „Rekayasa perangkat lunak‟, *Yogyakarta: Penerbit Andi*.

Mandala, E. P. W. (2015) „Web Programing Project 1: epwm Forum‟, *Yogyakarta: Andi*.

Shalahuddin, M. and Rosa, A. S. (2018) „Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek) Edisi Revisi‟, *Bandung, Penerbit Informatika*.

Susanto, C., Ari, M. and GA, M. (2013) „Penyakit kulit dan kelamin‟, *Yogyakarta: Nuha Medika*.

Sutojo, T., Mulyanto, E. and Suhartono, V. (2011) „Kecerdasan Buatan. Yogyakarta: Andi‟.

Tohari, H. (2014) „Analisa Serta Perancangan System Informasi Melalui Pendekatan

UML, Yogyakarta: CV‟, *Andi Offset*.

Abrori, A., Hermawan, A. D. and Inayati, S. (2016) „Faktor-Faktor Yang Berhubungan Dengan Terjadinya Servisitis Pada Wanita Di Lingkungan Keluarga Pegawai Negeri Sipil Pemerintah Provinsi Kalimantan Barat‟, *Unnes Journal of Public Health*, 5(3), p. 263. doi: 10.15294/ujph.v5i3.11274.

Anisah, A. and Kuswaya, K. (2017) „Analisis Dan Perancangan Sistem Informasi Pengolahan Data Pengeluaran, Penggunaan Bahan Dan Hutang Dalam Pelaksanaan Proyek Pada Pt Banamba Putratama‟, *Simetris : Jurnal Teknik Mesin, Elektro dan Ilmu Komputer*, 8(2), p. 507. doi: 10.24176/simet.v8i2.1352.

Destiningrum, M. and Adrian, Q. J. (2017) „Sistem Informasi Penjadwalan Dokter Berbassis Web Dengan Menggunakan Framework Codeigniter (Studi Kasus: Rumah Sakit Yukum Medical Centre)‟, *Jurnal Teknoinfo*, 11(2), p. 30. doi: 10.33365/jti.v11i2.24.

Dwi Harningtyas, C. (2017) „Pemberian Terapi Oral Untuk Pasien Uretritis Gonore Dengan Komplikasi Lokal Pada Pria: Laporan Kasus Administration of Oral Therapy For Gonorrheal Urethritis Patients With Local Complications In Men: Case Reports‟, *Journal of Agromedicine and Medical Sciences*, 3(3), pp. 1–6.

Firman, A. *et al.* (2016) „Sistem Informasi Perpustakaan Online Berbasis Web‟, *E- Journal Teknik Elektro Dan Komputer*, 5(2), pp. 29–36. doi: 10.35793/jtek.5.2.2016.11657.

Hendini, A. (2016) „Pemodelan UML sistem informasi monitoring penjualan dan stok barang (studi kasus: distro zhezha pontianak)‟, *Jurnal Khatulistiwa Informatika*, 4(2).

Manalu, L. and Harianja, A. P. (2018) „Artificial Intelligence (AI) Susun Angka Bentuk Kotak 4 X 4 Menggunakan Pencarian Heuristik Dengan Algoritma Bfs‟, *Jurnal Teknik Informatika UNIKA Santo Thomas*, 3(1), pp. 6–11.

Prayitno, A. (2015) „Pemanfaatan Sistem Informasi Perpustakaan Digital Berbasis Website Untuk Para Penulis‟, *IJSE-Indonesian Journal on Software Engineering*, 1(1).

Rozaq, A., Lestari, K. F. and Handayani, S. (2015) „Sistem iformasi produk dan data calon jamaah haji dan umroh pada Pt. Travellindo Lusiyana Banjarmasin berbasis web‟, *Jurnal POSITIF*, 1(1), pp. 1–13. doi: 10.1117/12.532107.

Suendri, S. (2019) „Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Remunerasi Dosen Dengan Database Oracle (Studi Kasus: UIN Sumatera Utara Medan)‟, *ALGORITMA: JURNAL ILMU KOMPUTER DAN INFORMATIKA*, 2(2), p. 1.

Wira, D., Putra, T. and Andriani, R. (2019) „Unified Modelling Language ( UML ) dalam Perancangan Sistem Informasi Permohonan Pembayaran Restitusi SPPD‟, 7(1).

Zm, A. P. and Erlansari, A. (2017) „Tiroid Menggunakan Metode‟, 5(3), pp. 270– 284.

Fricles Ariwisanto Sianturi (2019) „Analisa metode teorema bayes dalam mendiagnosa keguguran pada ibu hamil berdasarkan jenis makanan‟, *Teknik Informasi dan Komputer (Tekinkom)*, 2(1), pp. 87–92. Available at: [http://jurnal.murnisadar.ac.id/index.php/Tekinkom/article/view/78.](http://jurnal.murnisadar.ac.id/index.php/Tekinkom/article/view/78)

# Lampiran I. Biodata Ketua Tim Peneliti Ketua.

**DAFTAR RIWAYAT HIDUP**

1. **IDENTITAS DIRI**

|  |  |  |
| --- | --- | --- |
| 1 | Nama Lengkap (dengan gelar) | Raja Ayu Mahessya,  S.Kom, M.Kom |
| 2 | Jenis Kelamin | Perempuan |
| 3 | Jabatan Fungsional | Asisten Ahli |
| 4 | NIP/NIK/Identitas lainnya |  |
| 5 | NIDN | 1023119001 |
| 6 | Scopus ID. |  |
| 7 | Orcid ID. |  |
| 8 | ResearcherID. |  |
| 6 | Tempat dan Tanggal Lahir | Padang, 23 November 1990 |
| 7 | EMail | [ayumahessya@gmail.com](mailto:ayumahessya@gmail.com) |
| 8 | Nomor Teleon/HP | 085264454350 |
| 9 | Alamat Kantor | Kampus Universias Putra Indonesia “YPTK” Padang , Jl. Raya Lubuk Begalung, Padang, Sumatera Barat |
| 10 | WA / Sosmed | 085264454350 |
| 12 | Mata Kuliah yang Diampu | 1. Pemodelan dan Simulasi |
|  |  | 2. Matematika Diskrit |
|  |  | 3. Konsep Sistem Informasi |
|  |  | 4. Pemograman Berbasis Objek (OOP)  5. Bahasa Pemrograman III (JAVA) |

1. **RIWAYAT PENDIDIKAN**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **S-1** | **S-2** | **S-3** |
| Nama Perguruan Tinggi | Universitas Putra Indonesia YPTK | Universitas Putra Indonesia YPTK |  |
| Bidang Ilmu | Sistem Informasi | Sistem Informasi |  |
| Tahun Masuk-Lulus | 2008-2012 | 2012-2013 |  |
| Judul Skripsi/Tesis/ Disertasi | Sistem Informasi Pengolahan Data Siswa Grahita Pada UPTD Panti Sosial Bina Grahita Hararapan Ibu (PSBGHI) Dengan Menggunakan Bahasa Pemrograman Netbeans Java 6.9.1 | Jaringan Syaraf Tiruan Menggunakan Algoritma Backpropogation Untuk Memprediksi Kecerdasan Siswa (Studi Kasus : LP3I Course Center Padang) |  |
| Nama Pembimbing/ Promotor | 1. **Silfia Andini, S.Kom, M.Kom** 2. **Sri Rahmawati, S.Kom, M.Kom** | 1. **DR. Rusdiyanto Roestam** 2. DR. H. Sarjon Defit, S.Kom, M.Sc |  |

1. **PENGALAMAN PENELITIAN DALAM 5 TAHUN TERAKHIR**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Tahun | Judul Penelitian | Pendanaan | |
| Sumber | Jlh (Juta Rp) |
|  | 2014 | Memprediksi Kecerdasan Siswa Menggunakan Jaringan Syaraf Tiruan Berbasis Algoritma Backpropagation (Studi Kasus Di Lp3i Course Center Padang) | Mandiri |  |
|  | 2016 | Membangun Sistem Pakar Untuk Diagnosa Infeksi Virus Pada Anak Dengan Menggunakan Metode Forward Chaining | Dikti | Rp.11.500.000,- |
|  | 2016 | Sistem Informasi Pengolahan Data Siswa Grahita Menggunakan Bahasa Pemrograman Java Netbeans (Studi Kasus Di Uptd Panti Sosial Bina Grahita Harapan Ibu (PSBGHI)) | Mandiri |  |
|  | 2017 | Penerapan Queueing Theory Sistem Antrian Pendaftaran Bpjs Di Kota Bukittinggi Berbasis Web | Mandiri |  |
|  | 2017 | Pemodelan Dan Simulasi Sistem Antrian Pelayanan Pelanggan Menggunakan Metode Monte Carlo Pada Pt Pos Indonesia (Persero) Padang | Mandiri |  |
|  | 2017 | Aplikasi Arsip Digital Berbasis Web (Studi Kasus : Dinas Arsip Dan Perpustakaan Daerah Kota Sawahlunto) | Mandiri |  |
|  | 2018 | Pembangunan Aplikasi Pemasaran Perumahan Developer Kota Padang Dengan Konsep Customer Relationship Managemen (CRM) | Dikti | Rp.16.626.000 |
|  |  |  |  |  |

1. **PENGALAMAN PENGABDIAN KEPADA MASYARAKAT DALAM 5 TAHUN TERAKHIR**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Tahun | Judul Pengabdian Kepada Masyarakat | Pendanaan | |
| Sumber | Jlh (Juta Rp) |
|  | 2015 | Pelatihan Pemanfaatan Teknologi Informasi Dan Komunikasi Untuk Meningkatkan Akhlakul Kharimah Generasi Muda Islam Dalam Kegiatan Pesantren Ramadhan Pemerintah Kota Padang | UPI-YPTK | 10 |
|  | 2016 | Bakti Sosial Dan Sosialisasi Pengenalan Teknologi Informasi Bersama Ves Community Dan 1000 Guru Di Jorong Lambeh, Nagari Iv Koto, Kecamatan Palembayan, Kabupaten Agam, Propinsi Sumatera Barat | UPI-YPTK | 10 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

1. **PUBLIKASI ARTIKEL ILMIAH DALAM JURNAL DALAM 5 TAHUN TERAKHIR**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Judul Artikel Ilmiah | Nama Jurnal | Volume/ Nomor/Tahun |
| 1. | Memprediksi Kecerdasan Siswa Menggunakan Jaringan Syaraf Tiruan Berbasis Algoritma Backpropagation (Studi Kasus Di LP3I Course Center Padang) | Majalah KomTekInfo-YPTK, UPI YPTK Padang | ISSN : 2356-0010, Volume 1 No.2 Desember 2014 |
| 2. | Pengembangan Multimedia Interaktif Dalam Media Pembelajaran Mengenal Huruf Hijaiyah | J-Click STMIK-AMIK Jayanusa Padang | ISSN : 2355-7958 Volume 4 No 2 Maret 2016 |
|  | Membangun Sistem Pakar Untuk Diagnosa Infeksi Virus Pada Anak Dengan Menggunakan Metode Forward Chaining | Jurnal Teknologi, UPI YPTK Padang | ISSN:2301-4474 Vol.6 No.2 |
|  | Sistem Informasi Pengolahan Data Siswa Grahita Menggunakan Bahasa Pemrograman Java Netbeans (Studi Kasus Di Uptd Panti Sosial Bina Grahita Harapan Ibu (PSBGHI)) | LPPM Institut Teknologi Padang | ISSN:2338-2724 Vol 4 No 1 |
|  | Penerapan Queueing Theory Sistem Antrian Pendaftaran BPJS Di Kota Bukittinggi Berbasis Web | Ejurnal Kopertis 10 | E-ISSN: 2502-096X |
|  | Pemodelan Dan Simulasi Sistem Antrian Pelayanan Pelanggan Menggunakan Metode Monte Carlo Pada PT Pos Indonesia (Persero) Padang | Jurnal Hang tuah | E-ISSN: 2579 -3918 |
|  | Aplikasi Arsip Digital Berbasis Web (Studi Kasus : Dinas Arsip Dan Perpustakaan Daerah Kota Sawahlunto) | J-Click STMIK-AMIK Jayanusa Padang | E-ISSN:2541-2469 |
|  |  |  |  |

1. **PEMAKALAH SEMINAR ILMIAH (ORAL PRESENTATION) DALAM 5 TAHUN TERAKHIR**

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Nama Pertemuan Ilmiah/Seminar | Judul Artikel Ilmiah | Waktu dan Tempat |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |
| 5 |  |  |  |
| 6 |  |  |  |
| 7 |  |  |  |

1. **KARYA BUKU DALAM 5 TAHUN TERAKHIR**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Judul Buku | Tahun | Jumlah Halaman | Penerbit |
| 1 | Dasar Robotika | 2018 | 120 | Pustaka Galeri Mandiri |

1. **PEROLEHAN HKI DALAM 5-10 TAHUN TERAKHIR**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Judul/Tema HKI | Tahun | Jenis | Nomor P/ID |
| 1. |  |  |  |  |
| 2. |  |  |  |  |

1. **PENGALAMAN MERUMUSKAN KEBIJAKAN PUBLIK/REKAYASA SOSIAL LAINNYA DALAM 5 TAHUN TERAKHIR**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Judul/Tema /Jenis Rekayasa Sosial Lainnya yang Telah Diterapkan | Tahun | Tempat Penerapan | Respon Masyarakat |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**J. Penghargaan Dalam 10 Tahun Terakhir (dari pemerintah atau institusi lainnya)**

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Jenis Penghargaan | Institusi Pemberi Penghargaan | Tahun |
| 1. |  |  |  |
| 2. |  |  |  |

Semua data yang saya isikan dan tercantum dalam biodata ini adalah benar dan dapat dipertanggungjawabkan secara hukum Demikian biodata ini saya buat dengan sebenarnya untuk memenuhi salah satu persyaratan dalam pengajuan NIDN di Universitas Putra Indonesia “YPTK” Padang.

Padang, 6 November 2018

**Raja Ayu Mahessya**

**NIDN:1023119001**