Lembar Desain COVER BUKU

(desain cover harus original)

PADA COVER Harus Tercantum :

Nama Penulis (nama Pembimbing + Nama Mahasiswa)

+ Logo Penerbit Di bagian bawah

Cahyo Prianto

Resa Rianti

Nur Tri Ramadhanti Adiningrum

**PENDEKATAN MACHINE LEARNING DALAM PREDIKSI**

**KANKER PAYUDARA MENGGUNAKAN SUPPORT**

**VECTOR MACHINE**

**PENDEKATAN MACHINE LEARNING   
DALAM PREDIKSI KANKER PAYUDARA MENGGUNAKAN SUPPORT VECTOR MACHINE**

Cahyo Prianto

Resa Rianti

Nur Tri Ramadhanti Adiningrum

**CANTUMKAN**

**Logo Penerbit Disini**

**PENDEKATAN MACHINE LEARNING DALAM PREDIKSI KANKER PAYUDARA MENGGUNAKAN SUPPORT VECTOR MACHINE**

©TULISKAN NAMA PENERBIT DISINI

Penulis:

……..

Editor:

(Nama Penguji Sidang)

Cetakan Pertama: Isi dengan Bulan saat upload buku

Cover: Tim Penyusun

Tata Letak: Tim Kreatif Penerbit

Hak Cipta 2023, pada Penulis. Diterbitkan pertama kali oleh:

**ISI NAMA PENERBIT**

ISI ALAMAT PENERBIT

Website: [WEBSITE](http://www.rcipress.rcipublisher.org/) PENERBIT

E-mail: [EMAIL](mailto:rumahcemerlangindonesia@gmail.com) PENERBIT

Copyright © 2023 by NAMA PENERBIT

All Right Reserved

- Cet. I –: NAMA PENERBIT, TAHUN TERBIT

Dimensi : 14,8 x 21 cm

ISBN: KOSONGKAN DULU

Hak cipta dilindungi undang-undang

Dilarang memperbanyak buku ini dalam bentuk dan dengan cara apapun tanpa izin tertulis dari penulis dan penerbit

Undang-undang No.19 Tahun 2002 Tentang

**Hak Cipta Pasal 72**

Undang-undang No.19 Tahun 2002 Tentang Hak Cipta Pasal 72

Barang siapa dengan sengaja melanggar dan tanpa hak melakukan perbuatan sebagaimana dimaksud dalam pasal ayat (1) atau pasal 49 ayat (1) dan ayat (2) dipidana dengan pidana penjara masing-masing paling sedikit 1 (satu) bulan dan/atau denda paling sedikit Rp.1.000.000,00 (satu juta rupiah), atau pidana penjara paling lama 7 (tujuh) tahun dan/atau denda paling banyak Rp.5.000.000.000,00 (lima miliar rupiah).

Barang siapa dengan sengaja menyiarkan, memamerkan, mengedarkan, atau menjual kepada umum suatu ciptaan atau barang hasil pelanggaran hak cipta terkait sebagai dimaksud pada ayat (1) dipidana dengan pidana penjara paling lama 5 (lima) tahun dan/atau denda paling banyak Rp.500.000.000,00 (lima ratus juta rupiah).

# KATA PENGANTAR

Segala puji bagi Allah, Tuhan Yang Maha Esa atas rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan buku ini. Tak lupa juga mengucapkan salawat serta salam semoga senantiasa tercurahkan kepada Nabi Besar Muhammad SAW, karena berkat beliau, kita mampu keluar dari kegelapan menuju jalan yang lebih terang.

Kami ucapkan juga rasa terima kasih kami kepada pihak-pihak yang mendukung lancarnya buku ini mulai dari proses penulisan hingga proses cetak, yaitu orang tua kami, rekan-rekan kami, penerbit, dan masih banyak lagi yang tidak bisa kami sebutkan satu per satu.

Adapun, buku kami yang berjudul Bahan Ajar: Pendekatan machine learning dalam prediksi kanker payudara menggunakan support vector machine ini telah selesai kami buat secara semaksimal dan sebaik mungkin agar menjadi manfaat bagi pembaca yang membutuhkan informasi dan pengetahuan mengenai bagaimana memprediksi kanker payudara.

Dalam buku ini, tertulis bagaimana …..

yang menjadi alternatif pegangan bagi mahasiswa dan…

Kami sadar, masih banyak luput dan kekeliruan yang tentu saja jauh dari sempurna tentang buku ini. Oleh sebab itu, kami mohon agar pembaca memberi kritik dan juga saran terhadap karya buku ajar ini agar kami dapat terus meningkatkan kualitas buku.

Demikian buku ajar ini kami buat, dengan harapan agar pembaca dapat memahami informasi dan juga mendapatkan wawasan mengenai bidang sistem informasi manajemen serta dapat bermanfaat bagi masyarakat dalam arti luas. Terima kasih.

Bandung, 02 November 2022

**Penulis**

# DAFTAR ISI

[KATA PENGANTAR vii](#_Toc120884638)

[DAFTAR ISI viii](#_Toc120884639)

[DAFTAR GAMBAR x](#_Toc120884640)

[DAFTAR TABEL xii](#_Toc120884641)

[BAB 1 Apa Itu Kanker Payudara? 1](#_Toc120884642)

[1.1 Apa Itu Kanker? 1](#_Toc120884643)

[1.2 Apa Itu Payudara? 2](#_Toc120884644)

[1.3 Apa itu Kanker Payudara? 3](#_Toc120884645)

[1.4 Penyebab Kanker Payudara 3](#_Toc120884646)

[1.5 Gejala Kanker Payudara 4](#_Toc120884647)

[1.6 Cara Pengobatan Kanker Payudara 5](#_Toc120884648)

[1.7 Cara Mencegah Kanker Payudara 6](#_Toc120884649)

[1.8 Kasus Yang Pernah Terjadi 6](#_Toc120884650)

[BAB 2 Metodologi Pembuatan Aplikasi 8](#_Toc120884651)

[2.1 Algoritma Machine Learning Yang Digunakan 8](#_Toc120884652)

[2.1.1 Support Vector Machine 8](#_Toc120884653)

[2.2 Dataset 11](#_Toc120884654)

[2.2.1 Definisi Dataset 11](#_Toc120884655)

[2.2.2 Tipe Data 11](#_Toc120884656)

[2.2.3 Jenis Dataset 12](#_Toc120884657)

[2.2.4 Kualitas pada dataset 14](#_Toc120884658)

[2.2.5 Dataset Yang Digunakan 17](#_Toc120884659)

[2.3 Metodologi Pengolahan Data 22](#_Toc120884660)

[BAB 3 Apa Saja Perangkat Yang Digunakan Untuk Membuat Aplikasi? 24](#_Toc120884661)

[3.1 Bahasa Pemrograman Yang Digunakan 25](#_Toc120884662)

[3.1.1 Python 25](#_Toc120884663)

[3.1.2 HTML 26](#_Toc120884664)

[3.1.3 CSS 27](#_Toc120884665)

[3.1.4 Bootstrap 28](#_Toc120884666)

[3.1.5 Django 28](#_Toc120884667)

[3.2 Penyimpanan Yang Digunakan 29](#_Toc120884668)

[3.2.1 Database MySQL 29](#_Toc120884669)

[3.3 Perangkat Lunak Yang Dibutuhkan 31](#_Toc120884670)

[3.3.1 Visual Studio Code 31](#_Toc120884671)

[3.3.2 XAMPP 32](#_Toc120884672)

[3.3.3 Jupyter Notebook 33](#_Toc120884673)

[3.3.4 Tableau 35](#_Toc120884674)

[3.3.5 Anaconda 36](#_Toc120884675)

[3.3.6 Kaggle 36](#_Toc120884676)

[3.4 Spesifikasi Perangkat Keras 37](#_Toc120884677)

[DAFTAR PUSTAKA 39](#_Toc120884678)

# DAFTAR GAMBAR

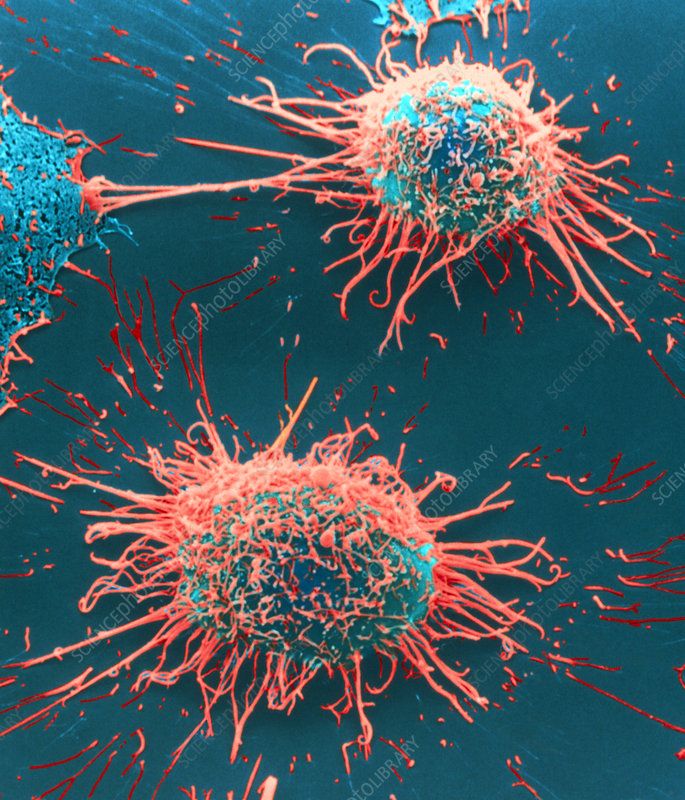
**2 |**

# DAFTAR TABEL

BUATLAH DAFTAR ISI DISINI

# BAB 1 Apa Itu Kanker Payudara?

## Apa Itu Kanker?

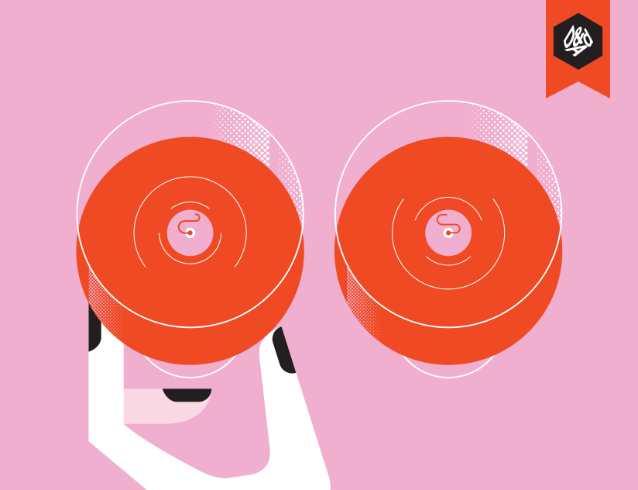


Kanker adalah sel yang tumbuh secara tidak terkendali dan tidak normal (abnormal) secara terus-menerus dan tidak terbatas. Pertumbuhan sel-sel kanker tidak terkoordinasi dengan jaringan lain dan dapat merusak tubuh. Kanker juga dikenal sebagai tumor ganas yang mengalami pertumbuhan abnormal yang tidak diketahui secara pasti penyebabnya. Sel normal hanya akan terus berkembang melalui proses pembelahan diri saat terjadi kematian atau kerusakan. Namun, sel kanker akan terus tumbuh meskipun tidak dibutuhkan oleh tubuh dan dapat merusak jaringan sel normal lainnya serta menyebar ke organ tubuh lain melalui jaringan ikat, darah, saraf, dan jaringan penunjang organ tubuh. Kanker adalah penyakit yang tidak menular yang timbul akibat kondisi fisik yang tidak normal dan pola hidup yang tidak sehat.[ ]

Gambar 1 Sel Kanker

Sumber:[*https://www.sciencephoto.com/media/294814/view/cervical-cancer-cells-sem*](https://www.sciencephoto.com/media/294814/view/cervical-cancer-cells-sem)

## Apa Itu Payudara?



Gambar 2 Breast Ilustration

Sumber :[https://www.dandad.org/awards/professional/2018/press-advertising-campaigns-local/](https://www.dandad.org/awards/professional/2018/press-advertising-campaigns-local/27113/coming-of-age)

Payudara, juga dikenal dengan nama mammae, adalah organ tubuh bagian atas dada pada spesies mamalia betina, termasuk perempuan manusia. Biasanya, payudara digunakan untuk menyalurkan air susu (ASI) pada bayi mamalia atau manusia yang baru lahir. Payudara memiliki nilai estetika karena bentuknya yang menonjol di bagian atas tubuh, membuat tubuh wanita terlihat lebih menarik. Fungsi utama payudara adalah sebagai produsen ASI dari lobulus-lobulus, yaitu kelenjar yang menghasilkan ASI, dan tubulus atau ductus yang mengantarkan ASI dari kelenjar ke putting susu (nipple). Kelenjar mammae merupakan ciri khas dari semua mamalia. Bentuknya mirip kerucut, namun tidak semua wanita memiliki ukuran yang sama. Jadi, payudara merupakan organ vital yang memproduksi ASI dalam jumlah besar.[ ]

## Apa itu Kanker Payudara?

Kanker payudara adalah kanker yang paling banyak ditemukan pada wanita dan angka kematiannya masih menempati posisi kedua diantara kanker lainnya[1]. Carsinoma Mammae, lebih dikenal sebagai kanker payudara, adalah penyakit di mana sel-sel kanker terbentuk di jaringan payudara. Kanker dapat muncul di kelenjar yang menghasilkan susu (lobulus), saluran yang membawa susu dari kelenjar ke puting (duktus), atau di jaringan lemak atau ikat di dalam payudara. Kanker payudara adalah penyebab kematian kedua pada wanita dan dapat menyebabkan gejala psikologis seperti kecemasan, depresi, panik, dan gegar otak pada pasiennya. Penyebab pasti dari kanker payudara masih belum diketahui, namun dianggap multifaktorial. Upaya pencegahan seperti pendidikan masyarakat dan remediasi diperlukan untuk menangani masalah ini, sesuai dengan kebutuhan individu. Mencegah kanker payudara sejak dini melalui SADARI (Pemeriksaan Payudara Sendiri) adalah hal yang penting yang harus dilakukan oleh setiap orang. Namun, jika kanker payudara terdiagnosis, pemeriksaan lebih lanjut oleh dokter diperlukan. [3]

Gambar 3 Breast Cancer illustration

Sumber : [https://www.freepik.com/premium-vector/cancer-design-illustration\_3](https://www.freepik.com/premium-vector/cancer-design-illustration_3801675.htm?epik=dj0yJnU9ejBEd2lLWVBZVmU4dTJoOTVpU2l6SnllYUY3bF9nTGQmcD0wJm49YUxRaF8xdHFQNk9yd3F1eVpWWmg4dyZ0PUFBQUFBR09pNWFr)

## Penyebab Kanker Payudara

Faktor penyebab yang memungkinkan untuk timbulnya kanker payudara antara lain :

1. Mengkonsumsi makanan yang mengandung lemak berlebih dan berprotein tinggi, tetapi serat terlalu banyak. Dengan mengkonsumsi makanan seperti itu mengandung zat karsinogen yang dapat merangsang pertumbuhan sel kanker.
2. Pada usia muda telah mengkonsumsi pil kontrasepsi. Penelitian telah membuktikan jika usia dini (remaja) sudah meminum oral (pil) sangat tinggi risikonya terserang kanker payudara
3. Keturunan dari anggota keluarga yang pernah terkena kanker payudara.
4. Mengkonsumsi alkohol terlalu banyak.
5. Wanita merokok memiliki risiko paling besar dapat terserang kanker payudara karena kontaminasi senyawa kimia yang terlalu berlebihan

## Gejala Kanker Payudara

Pada stadium awal, penyakit ini tidak dapat menunjukkan gejala tertentu. Oleh karena itu, sangat penting untuk melakukan pemeriksaan payudara sendiri setiap bulan, sepuluh hari setelah akhir periode perempuan. Rasakan rabaan searah jarum jam dengan hati-hati untuk mendeteksi adanya benjolan atau perubahan pada payudara.

Beberapa gejala kanker payudara yang bisa dialami oleh penderitanya adalah:

1. Benjolan atau pengerasan payudara yang berbeda dari jaringan di sekitarnya.
2. Darah keluar dari puting.
3. Kemerahan atau penggunaan pori-pori kulit payudara yang menyerupai kulit jeruk.
4. Nyeri dan bengkak pada payudara.
5. Pengelupasan kulit di sekitar puting.
6. Perubahan pada kulit payudara, seperti depresi.
7. Perubahan ukuran, bentuk, atau tampilan payudara.
8. Puting ditarik (ditarik atau dibalik) ke dalam.
9. Benjolan atau pembengkakan di ketiak bagian bawah.

Ada banyak penyebab munculnya benjolan di payudara, yang sebagian besar bukan kanker. 90% dari massa payudara tidak bersifat kanker. Gangguan payudara non-kanker termasuk massa jinak seperti fibroadenoma dan kista serta infeksi.

## Cara Pengobatan Kanker Payudara

Perawatan untuk penyakit ini tergantung pada jenis kanker yang Anda derita, stadium kanker, ukuran massa, dan sensitivitas sel kanker terhadap hormon. Pengobatan kanker payudara bisa sangat efektif, mencapai probabilitas kelangsungan hidup 90% atau lebih tinggi, terutama ketika penyakit ini diidentifikasi lebih awal.  Dokter akan menentukan terapi yang paling tepat untuk kondisi pasien berdasarkan hal-hal tersebut. Ada beberapa pilihan pengobatan untuk kanker payudara, antara lain:

1. Pembedahan, yang meliputi pengangkatan kanker atau benjolan (lumpektomi), pengangkatan seluruh payudara (mastektomi), pengangkatan sejumlah kelenjar getah bening (biopsi simpul sentinel), atau pengangkatan beberapa kelenjar getah bening (diseksi kelenjar getah bening aksila).
2. Radioterapi, yang menggunakan sinar-X dan proton untuk membunuh sel kanker.
3. Kemoterapi, yaitu dilakukan dengan menggunakan obat-obatan tertentu untuk membunuh sel kanker.
4. Terapi hormonal, untuk mencegah sel kanker mendapatkan hormon yang mereka butuhkan untuk tumbuh.
5. Terapi biologis, yang bekerja dengan sistem kekebalan pasien untuk membantu melawan sel kanker, atau untuk mengontrol efek samping dari perawatan kanker lainnya.
6. Terapi radiasi menggunakan sinar berenergi tinggi untuk membunuh sel kanker.

## Cara Mencegah Kanker Payudara

Kanker payudara memang sangat mengerikan. Penyakit ini bisa disebabkan oleh banyak faktor seperti faktor keturunan, faktor lingkungan, dan gaya hidup yang tidak sehat. Untuk mencegah kanker payudara, sebaiknya periksa kondisi payudara secara rutin dengan cara menyentuh dan memijatnya. Cara ini cukup efektif untuk mendeteksi pertumbuhan sel kanker payudara sebelum mencapai stadium lanjut. Kanker ini juga dapat dicegah dengan cara-cara yang dijelaskan di bawah ini.

1. Berolahraga minimal 30 menit sehari
2. Berhenti merokok
3. Konsumsi buah dan sayur secara teratur
4. Konsumsi produk kedelai
5. Asupan serat makanan
6. Hindari alkohol
7. Kurangi asupan gula
8. Hindari konsumsi lemak jahat
9. Mengetahui riwayat orang tua
10. Hindari memakan daging merah
11. Mengkonsumsi obat-obatan
12. Operasi

## Kasus Yang Pernah Terjadi

Diperkirakan ada 18,1 juta kasus kanker di seluruh dunia pada tahun 2020. Dari jumlah tersebut, 9,3 juta kasus terjadi pada pria dan 8,8 juta pada wanita. Kanker yang paling umum secara global adalah kanker payudara dengan total 2.261.419 kasus atau sebanyak 12.5% dari total persentase berbagai kanker dan merupakan kanker yang menempati posisi pertama kasus kanker terbayak di dunia.

Pada tahun 2020, terdapat 2,3 juta wanita yang didiagnosis menderita kanker payudara dan 685.000 kematian secara global. Pada akhir tahun 2020, terdapat 7,8 juta wanita hidup yang didiagnosis menderita kanker payudara dalam 5 tahun terakhir, kanker paling umum di dunia. Ada lebih banyak kehidupan yang berakhir akibat oleh wanita karena kanker payudara secara global daripada jenis kanker lainnya. Kanker payudara terjadi di setiap negara di dunia pada wanita pada usia berapa pun setelah pubertas tetapi dengan tingkat yang meningkat di kemudian hari.

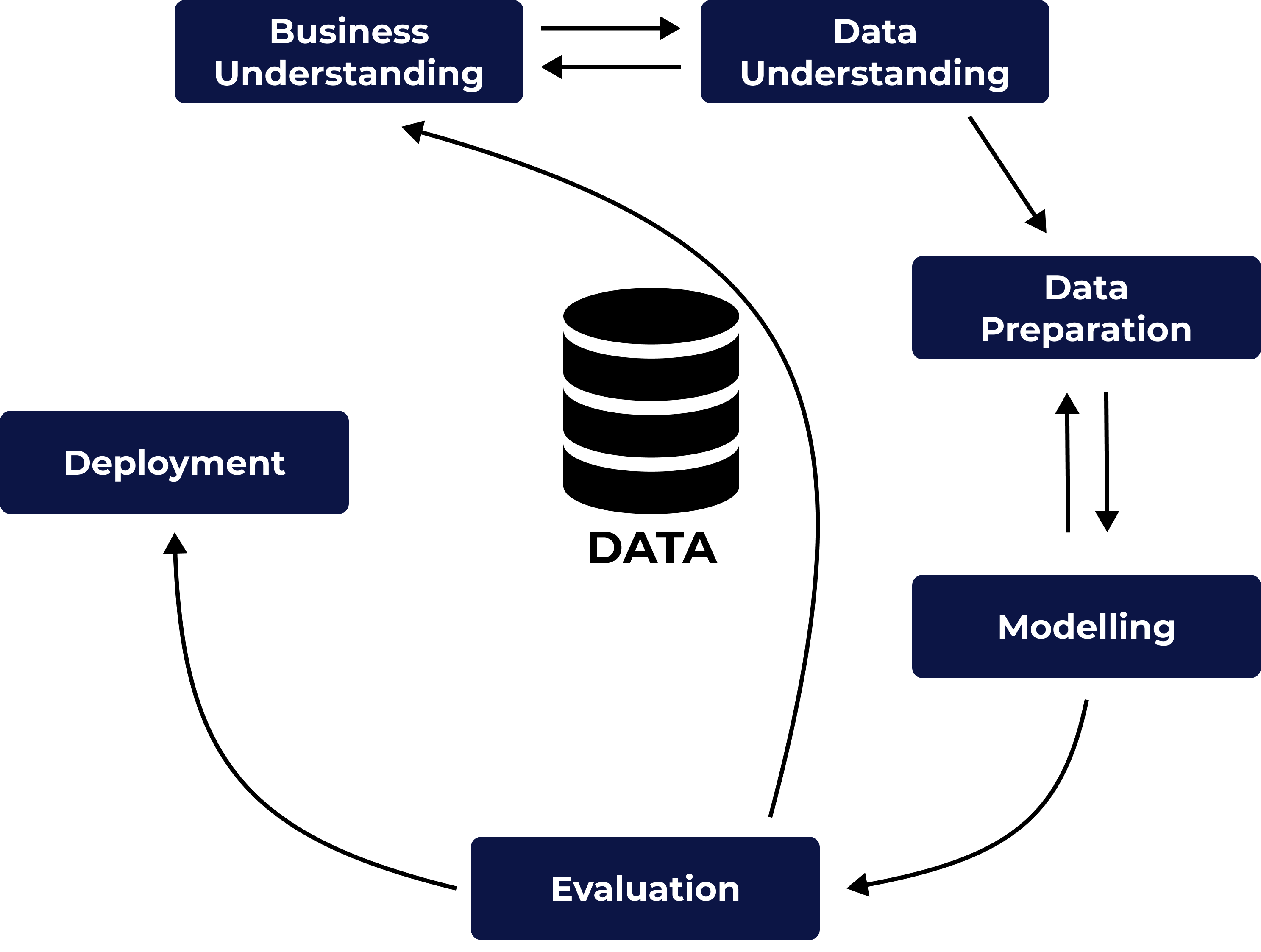
Kelangsungan hidup kanker payudara selama minimal 5 tahun setelah diagnosis berkisar dari lebih dari 90% di negara-negara berpenghasilan tinggi, hingga 66% di India dan 40% di Afrika Selatan.

Deteksi dan pengobatan dini telah terbukti berhasil di banyak negara maju dan harus diterapkan di negara-negara dengan sumber daya terbatas di mana beberapa alat standar tersedia. Angka kematian akibat kanker payudara standar usia di negara-negara berpenghasilan tinggi turun 40% antara tahun 1980-an dan 2020. Negara-negara yang berhasil menurunkan angka kematian akibat kanker payudara telah mampu mencapai penurunan angka kematian akibat kanker payudara tahunan sebesar 2-4% per tahun. Jika penurunan kematian tahunan sebesar 2,5% per tahun terjadi di seluruh dunia, 2,5 juta kematian akibat kanker payudara akan dapat dihindari antara tahun 2020 dan 2040.

# BAB 2 Metodologi Pembuatan Aplikasi

## Metodologi Pengolahan Data

Diagram alur metodologi pengolahan data yang digunakan ialah menggunakan metodologi CRISP-DM (*Cross Industry Standard Process for Data Mining*). Metodologi penelitian CRISP-DM digunakan dalam penelitian karena metodologi ini mampu menjabarkan setiap langkah pengolahan data dengan jelas serta mudah dipahami untuk diimplementasikan.



Gambar 4 CRISP-DM

Berdasarkan diagram alur metodologi penelitian diatas, dapat dipaparkan sebagai berikut :

## *Business* *Understanding*

Pada tahap ini merupakan tahap untuk menentukan seperti apa proses bisnis yang akan dibangun untuk melakukan hal apa yang ingin dicapai. Selain itu, dalam tahap ini juga diperlukan analisis apa saja alat dan bahan yang akan diperlukan untuk setiap fase kegiatan.

Dalam hal ini, tujuan kegiatan yang dilakukan adalah memprediksi kanker payudara berdasarkan faktor-faktor spesifik. Prediksi kanker tersebut akan menggunakan data kanker payudara. Pendekatan yang dilakukan dalam memprediksi kanker payudara akan dilakukan menggunakan *machine learning*.

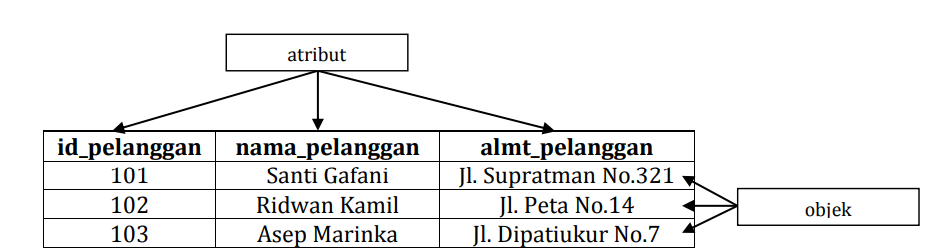
Setelah melakukan prediksi kanker payudara menggunakan *machine learning*, model prediksi akan diterapkan dalam bentuk *web base*.

Alat yang digunakan adalah berupa perangkat keras berupa laptop. Perangkat lunak yang digunakan dalam pengolahan data kanker untuk prediksi ialah Jupyter Notebook dan Anaconda. Perangkat lunak yang digunakan dalam pembangunan web base adalah Visual Studio Code, XAMPP, dan Tableu.

## *Data Understanding*

Pada tahap *data understanding*, hal yang dilakukan adalah melakukan pemahaman bisnis yang berfokus untuk mengidentifikasi, mengumpulkan dan menganalisis kumpulan data untuk mencapai tujuan yang dipaparkan pada *Business* *Understanding*.

Definisi set data (dataset/himpunan data) merupakan kumpulan objek dan atributnya. Nama lain dari objek yang sering digunakan diantaranya record, point, vector, pattern, event, observation, case, sample, instance, entitas. Objek digambarkan dengan sejumlah atribut yang menerangkan sifat atau karakteristik dari objek tersebut. Atribut juga sering disebut variabel, field, fitur, atau dimensi. Atribut adalah sifat/properti/karakteristik objek yang nilainya bisa bermacam-macam dari satu objek dengan objek lainnya, dari satu waktu ke waktu yang lainnya. Sebagai contoh seorang pelanggan merupakan objek, dimana objek pelanggan tersebut memiliki beberapa atribut seperti id pelanggan, nama, alamat dan lain-lain. Setiap pelanggan memungkinkan memiliki nilai atribut yang berbeda dengan pelanggan lainnya, serta memungkinkan perubahan nilai atribut dari waktu ke waktu. [ ]



Berikut adalah berbagai pemaparan mengenai data.

### **Tipe Data**

Tipe atribut dapat dibedakan dari nilai beserta sifatnya. Ada empat sifat yang dimiliki atribut secara umum, yaitu:

* + 1. Pembeda (distinctness): = dan ≠
    2. Urutan (order): <, >, ≤, ≥
    3. Penjumlahan, Pengurangan (addition): + dan –
    4. Perkalian, Pembagian (multiplication): \* dan /

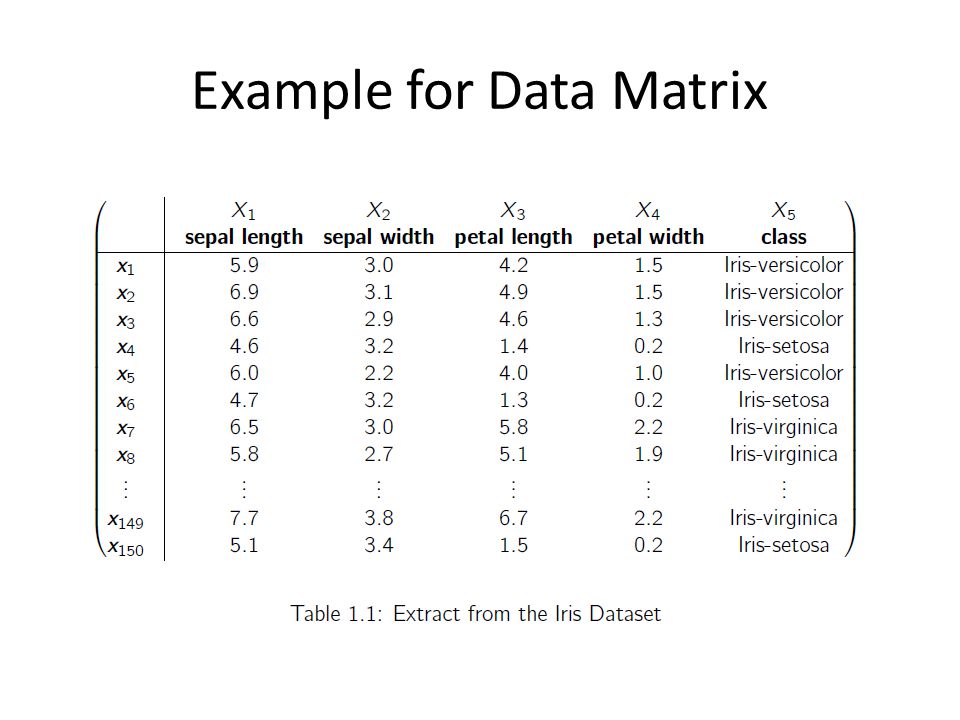
Umumnya tipe atribut ini ada dua yaitu kategori (kualitatif) dan numerik (kuantitatif). Dari kedua tipe tersebut dibagi lagi menjadi beberapa sub tipe yang disesuaikan dengan sifat nilai yang dimilikinya.

### **Jenis Dataset**

Jenis data dapat dibedakan menjadi tiga kelompok, yaitu data record, data berbasis grafik (graph data), dan data terurut (ordered data). Pada perkuliahan ini kita hanya akan menggunakan data record untuk proses data mining. Kebanyakan metode data mining mengasumsikan bahwa set data yang diproses adalah kumpulan baris data (record/entries/objects), dimana setiap barisnya terdiri atas sejumlah fitur (atribut) yang tetap. Dalam set data berbentuk data record, tidak ada hubungan antara baris data dengan baris data yang satu dengan baris data yang lainnya dan juga dengan set data yang lain. Setiap baris data berdiri sendiri sebagai sebuah data individu. Dalam sistem basis data, umumnya ada sejumlah tabel yang saling berhubungan menggunakan suatu kunci (kunci utama, kunci tamu). Akan tetapi dalam set data record, diasumsikan bahwa hanya ada satu tabel yang berisi sejumlah baris data. Oleh karena itu, biasanya set data yang diolah dalam data mining adalah keluaran dari sistem data warehouse yang menggunakan query untuk melakukan pengambilan data dari sejumlah tabel dalam sistem basis data. Ada beberapa contoh set data yang masuk dalam jenis data record, diantaranya data matrik (matrix data), data keranjang belanja (market basket data), dan data dokumen.

1. Data Matrik

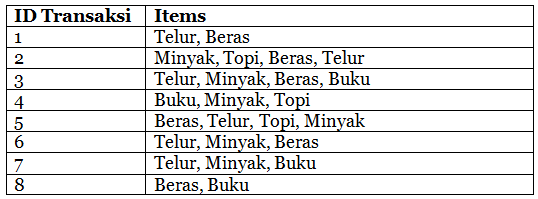
Jika set data berisi kumpulan data yang mempunyai sejumlah atribut (fitur) numerik yang sama, set data tersebut dapat dipandang sebagai vektor (data) dalam wilayah multidimensi, dimana masing-masing dimensi menyatakan satu atribut yang berbeda. Sekumpulan data matrik dapat diinterpretasikan sebagai mantrik M x N, dimana M adalah jumlah baris (satu baris menyatakan satu record/objek) dan N adalah jumlah kolom (dimana satu kolom menyatakan satu atribut/fitur). Contoh data matrik dapat dilihat pada gambar dibawah ini.



Gambar 7 Contoh Data Matrix

1. Data Keranjang Belanja/Transaksi

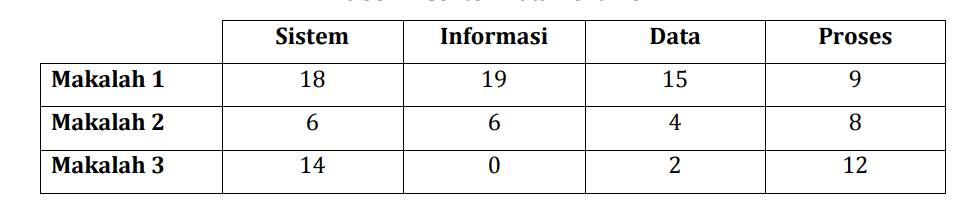
Data keranjang belanja (data transaksi) adalah set data yang setiap recordnya berisi sejumlah item dan jumlah item untuk sebuah transaksi bisa berbeda dengan transaksi yang lain. Contohnya bisa dilihat pada kasus keranjang belanja di pasar atau supermarket, dimana setiap pembeli melakukan pembelian barang yang jumlah dan jenisnya bisa berbeda dengan pembeli yang lain. Contoh data transaksi dapat dilihat pada tabel dibawah ini.



Gambar 8 Data Transaksi

1. Data Dokumen

Setiap dokumen merupakan satu vektor ‘term’. Tiap term merupakan satu komponen (atribut) dari vektor tersebut. Nilai dari setiap komponen menyatakan berapa kali kemunculan term tersebut dalam suatu dokumen. Contoh dari data dokumen dapat dilihat pada gambar dibawah ini.



### **Kualitas pada dataset**

Permasalahan kualitas data ditinjau dari aspek pengukuran data dan pengumpulan data.

* 1. Kesalahan Pengukuran (Measurement Error)

Kesalahan ini mengacu pada permasalahan hasil dari proses pengukuran. Problem yang umum terjadi adalah nilai yang dicatat berbeda dari nilai sebenarnya untuk beberapa tingkat. Pada atribut kontinu, beda numerik dari hasil pengukuran dengan nilai sebenarnya disebut dengan error. Yang termasuk dalam jenis kesalahan pengukuran adalah

1. Noise

Merupakan komponen random dari suatu error pengukuran. Noise berkaitan dengan modifikasi dari nilai asli. Contoh: distorsi atau penyimpangan dari suara orang saat berbicara di telepon yang jaringannya buruk.

1. Bias

Suatu variasi pengukuran dari kuantitas yang sedang diukur dengan pengurangan antara mean dan nilai kuantitas yang diketahui.

1. Precision

Kedekatan dari pengukuran berulang (dari kuantitas yang sama) satu dengan yang lainnya. Diukur dengan standar deviasi.

1. Accuracy

Kedekatan pengukuran terhadap nilai sebenarnya dari kuantitas yang sedang diukur.

* 1. Kesalahan Pengumpulan (Collection Error)

Mengacu pada kesalahan-kesalahan (error) seperti hilangnya objek data atau nilai atribut, atau lingkup objek data yang tidak tepat. Yang temasuk dalam kesalahan pengumpulan diantaranya:

1. Outliers

Merupakan objek data dengan sifat yang berbeda sekali dari kebanyakan objek data dalam data-set. Misalkan, terdapat data penelitian tentang tinggi anak siswa SMA yakni 160cm sampai 180cm. Tetapi dalam data tersebut terdapat anak yang mempunyai tinggi 140cm. Data anak dengan tinggi 140cm tersebut yang disebut data outlier, karena berbeda sangat jelas.

Terdapat beberapa hal yang mempengaruhi munculnya data outlier antara lain:

* + 1. Kesalahan dalam pemasukan data
    2. Kesalahan dalam pengambilan sample
    3. Memang ada data-data ekstrim yang tidak dapat dihindarkan keberadaannya.

1. Missing value

Merupakan nilai dari suatu atribut yang tidak ditemukan. Alasan terjadinya missing value adalah:

* + 1. Informasi tidak diperoleh (misal, orang-orang menolak untuk memberikan data umur dan berat bedan)
    2. Atribut yang mungkin tidak bisa diterapkan ke semua kasus (misal, pendapatan tahunan tidak bisa diterapkan pada seseorang yang pengangguran)

Penanganan missing values adalah dengan:

* + 1. Mengurangi objek data
    2. Memperkirakan missing values
    3. Mengabaikan missing values pada saat analisis
    4. Mengganti dengan semua nilai yang mungkin (tergantung probabilitasnya)

1. Duplicate data

Set data mungkin terdiri dari objek data yang ganda (duplikat), atau hampir selalu terjadi duplikasi antara satu dengan yang lainnya. Persoalan utama ketika menggabungkan data dari sumber-sumber yang bervariasi (heterogen).

Contoh: orang yang sama dengan alamat email yang lebih dari satu. Pembersihan data (data cleaning) merupakan proses yang berkaitan dengan permasalahan data yang duplikat. [ ]

Dalam hal ini, proses yang dilakukan ialah mengumpulkan data awal yang berasal dari Kaggle. Setelah data ditemukan, kemudian hal yang dilakukan adalah menjelajahi data dan melakukan verifikasi kualitas data (seberapa bersih atau kotornya data yang digunakan).

### **Dataset Yang Digunakan**

Dalam proses melakukan pengolahan data ini, digunakan suatu material yang dapat mendukung pengolahan data yang dilakukan. Material yang dimaksud ialah berupa data yang dapat digunakan pada proses pengolahan data.

Data yang digunakan dalam penelitian merupakan data pasien kanker payudara. Data tersebut didapatkan melalui situs bernama

Kaggle yang dapat dikunjungi melalui QR Code yang dicantumkan sebagai berikut:



Data bersifat berlabel dan memiliki jumlah kolom sebanyak 32 kolom dan jumlah total record sebanyak 569 *record*.

Kolom tersebut diantaranya adalah sebagai berikut.

Table 1 Atribut dan Arti dari Dataset

|  |  |
| --- | --- |
| Atribut | Arti |
| id | ID unik |
| diagnosis | Diagnosis masalah (M = *malignant*/ganas, B = *benign*/jinak) |
| Radius\_mean | Jari-jari lobus/kelenjar (rata-rata jarak dari pusat ke titik-titik di perimeter)  Perimeter adalah garis keliling inti sel yang diukur sebagai jumlah dari jarak. |
| Texture\_mean | Rata-rata Tekstur Permukaan (standar deviasi nilai *grayscale*) |
| Perimeter\_mean | Perimeter Luar Lobus (ukuran rata-rata tumor inti) |
| Area\_mean | Rata-rata Luas Lobus |
| Smoothness\_mean | Rata-rata Tingkat Kehalusan (rata-rata variasi lokal dalam panjang radius) |
| Compactness\_mean | Rata-rata dari rasio volume dan luas permukaan tumor (rata-rata dari perimeter2 / area - 1.0) |
| Concavity\_mean | Rata-rata keparahan bagian cekung dari kontur.  Sebuah cekungan di bagian bawah toraks (dada) yang disebabkan oleh perpindahan tulang rawan xiphoid ke belakang. |
| Concave ponts\_mean | Rata-rata untuk jumlah bagian cekung dari kontur |
| Symmetry\_mean | Rata-rata kesimetrisan.  Kesimetrisan mungkin terkait dengan penyakit payudara, masalah perkembangan, masalah hormonal, sindrom yang tidak diketahui, cedera atau periode perubahan tubuh yang signifikan seperti kehamilan atau menyusui.  Asimetri payudara mengacu pada ketika satu payudara memiliki ukuran atau bentuk yang berbeda dari yang lain. |
| Fractal\_dimension\_mean | Rata-rata dimensi fraktal (rata-rata "coastline approximation" - 1)  Analisis fraktal dari gambar spesimen jaringan payudara memberikan deskripsi numerik pola pertumbuhan tumor sebagai angka kontinu antara 1 dan 2. Angka ini, dimensi fraktal, adalah ukuran yang objektif dan dapat direproduksi dari kompleksitas arsitektur jaringan spesimen biopsi. |
| Radius\_se | Jari-jari lobus/kelenjar (Standard Error dari pusat ke titik-titik di perimeter)  Perimeter adalah garis keliling inti sel yang diukur sebagai jumlah dari jarak. |
| Texture\_se | Standard Error Tekstur Permukaan (standar deviasi nilai *grayscale*) |
| Perimeter\_se | Standard Error Perimeter Luar Lobus (ukuran Standard Error tumor inti) |
| Area\_se | Standard Error Luas Lobus |
| Smoothness\_se | Standard Error Tingkat Kehalusan (Standard Error variasi lokal dalam panjang radius) |
| Compactness\_se | Standard Error dari rasio volume dan luas permukaan tumor (Standard Error dari perimeter2 / area - 1.0) |
| Concavity\_se | Standard Error keparahan bagian cekung dari kontur.  Sebuah cekungan di bagian bawah toraks (dada) yang disebabkan oleh perpindahan tulang rawan xiphoid ke belakang. |
| Concave points\_se | Standard Error untuk jumlah bagian cekung dari kontur |
| Symmetry\_se | Standard Error kesimetrisan.  Kesimetrisan mungkin terkait dengan penyakit payudara, masalah perkembangan, masalah hormonal, sindrom yang tidak diketahui, cedera atau periode perubahan tubuh yang signifikan seperti kehamilan atau menyusui.  Asimetri payudara mengacu pada ketika satu payudara memiliki ukuran atau bentuk yang berbeda dari yang lain. |
| Fractal\_dimension\_se | Standard Error dimensi fraktal (Standard Error "coastline approximation" - 1)  Analisis fraktal dari gambar spesimen jaringan payudara memberikan deskripsi numerik pola pertumbuhan tumor sebagai angka kontinu antara 1 dan 2. Angka ini, dimensi fraktal, adalah ukuran yang objektif dan dapat direproduksi dari kompleksitas arsitektur jaringan spesimen biopsi. |
| Radius\_worst | Keparahan jari-jari lobus/kelenjar (keparahan jarak dari pusat ke titik-titik di perimeter)  Perimeter adalah garis keliling inti sel yang diukur sebagai jumlah dari jarak. |
| Texture\_worst | Keparahan Tekstur Permukaan (standar deviasi nilai *grayscale*) |
| Perimeter\_worst | Perimeter Luar Lobus (ukuran keparahan tumor inti) |
| Area\_worst | Keparahan Luas Lobus |
| Smoothness\_worst | Keparahan Tingkat Kehalusan (keparahan variasi lokal dalam panjang radius) |
| Compactness\_worst | Keparahan dari rasio volume dan luas permukaan tumor (keparahan dari perimeter2 / area - 1.0) |
| Concavity\_worst | Keparahan bagian cekung dari kontur.  Sebuah cekungan di bagian bawah toraks (dada) yang disebabkan oleh perpindahan tulang rawan xiphoid ke belakang. |
| Concave points\_worst | Keparahan untuk jumlah bagian cekung dari kontur |
| Symmetry\_worst | Keparahan kesimetrisan.  Kesimetrisan mungkin terkait dengan penyakit payudara, masalah perkembangan, masalah hormonal, sindrom yang tidak diketahui, cedera atau periode perubahan tubuh yang signifikan seperti kehamilan atau menyusui.  Asimetri payudara mengacu pada ketika satu payudara memiliki ukuran atau bentuk yang berbeda dari yang lain. |
| Fractal\_dimension\_worst | Keparahan dimensi fraktal (rata-rata "coastline approximation" - 1)  Analisis fraktal dari gambar spesimen jaringan payudara memberikan deskripsi numerik pola pertumbuhan tumor sebagai angka kontinu antara 1 dan 2. Angka ini, dimensi fraktal, adalah ukuran yang objektif dan dapat direproduksi dari kompleksitas arsitektur jaringan spesimen biopsi. |

Berdasarkan data tersebut, berikut adalah pemaparan kode python pada Jupyter Notebook dalam tahap *Data Understanding*.

|  |  |
| --- | --- |
| [1] |  |
| [2] |  |
| [3] |  |
| [4] |  |
| [5] |  |

Keterangan :

* 1. Baris 1 : Mengimport seluruh library yang dibutuhkan.
  2. Baris 2 : Mengimport data kedalam variabel dengan nama “df“. Kemudian menampilkan 5 data teratas.
  3. Baris 3 : Menampilkan informasi detail tabel dan data.
  4. Baris 4 : Menampilkan kolom apa saja yang terdapat di dalam data.
  5. Baris 5 : Menghapus atribut “id“, karena tidak akan terpakai.

## *Data Preparation*

Tahap *data preparation* dilakukan untuk menyiapkan data akhir untuk dilakukannya pemodelan dengan cara memperbaiki masalah yang terdapat pada data. Hal yang dilakukan dalam fase ini adalah membersihkan data, serta memilih data yang akan digunakan dalam pemodelan. Sehingga nantinya akan didapatkan data akhir yang siap dimodelkan.

Adapun tahap *data preparation* pada fase ini adalah sebagai berikut.

|  |  |
| --- | --- |
| [6] |  |
| [7] |  |
| [8] |  |
| [9] |  |
| [10] |  |
| [11] |  |
| [12] |  |
| [13] |  |
| [14] |  |
| [15] |  |
| [16] |  |

Keterangan :

* 1. Baris 6 : Menampilkan apakah terdapat data kosong pada suatu atribut atau tidak.
  2. Baris 7 : Membuat dan menampilkan korelasi antar atribut menggunakan heatmap correlation yang berasal dari library seaborn.
  3. Baris 8 : Menampilkan hasil perhitungan dari statistika berupa count, mean, std, min, 25%, 50%, 75%, max.
  4. Baris 9 : Membuat dan menampilkan Bar Chart dari atribut “diagnosis“ untuk mengetahui jumlah antara Malignant (M) dan Benign (B).
  5. Baris 10 : Menghapus atribut yang tidak terpakai, karena tidak memiliki korelasi antar atribut yang kuat
  6. Baris 11 : Mengubah seluruh nama atribut menjadi nama yang lebih mudah dipahami
  7. Baris 12 : Mengubah skala atribut menggunakan library StandardScaler dan preprocessing. Atribut “diagnosis“ memiliki nilai yang berubah menjadi angka (M menjadi 1, B menjadi 0).
  8. Baris 13 : Membuat dan menampilkan boxplot untuk melihat outlier pada data.
  9. Baris 14 : Membuat dan menampilkan boxplot untuk melihat outlier pada data dan menampilkan hasil outlier dengan array.
  10. Baris 15 : Menghitung nilai Q1 dan Q3 untuk menampilkan hasil Kuartil 1, Kuartil 2, dan nilai atribut (IQR)
  11. Baris 16 : menampilkan hasil IQR maksimum dan minimum

## *Modelling*

Tahap *modelling* atau pemodelan adalah menentukan algoritma teknik pemodelan yang akan digunakan untuk menyelesaikan masalah yang ada. Dalam fase ini, model yang digunakan adalah menggunakan model *Support Vector Machine*.

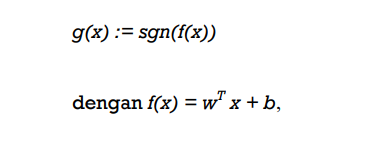
Karena model ini adalah model yang cocok untuk digunakan dalam melakukan prediksi terutama jika variabel yang mempengaruhi sangat banyak.

### **Support Vector Machine**

SVM adalah suatu aturan algoritma pembelajaran mesin yang diawasi digunakan untuk setiap klasifikasi dan tantangan regresi. Dimanfaatkan dalam masalah klasifikasi, dalam algoritma SVM, mengarah untuk memplot setiap item pengetahuan sebagai batas tertentu dalam ruang n-dimensi (di mana n adalah banyak fitur yang dimiliki) dengan nilai setiap fitur menjadi nilai koordinat tertentu. Selanjutnya, kami mengkategorikan dengan menemukan level super yang membedakan 2 kategori (kelas) dengan sangat baik. Nilai default cenderung digunakan untuk semua kunci parameter kecuali parameter kernel yang dipilih linier, dan parameter random\_state kami menetapkan nilainya sampai 40, untuk berhasil dalam hasil yang diperlukan [ ] .

Teknik ini menarik orang dalam bidang data mining maupun machine learning karena performansinya yang meyakinkan dalam memprediksi kelas suatu data baru.

Kita akan memulai pembahasan dengan kasus klasifikasi yang secara linier bisa dipisahkan. Dalam hal ini fungsi pemisah yang kita cari adalah fungsi linier. Fungsi ini bisa didefinisikan sebagai :



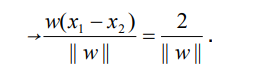
x, w Rn and bR. Masalah klasifikasi ini bisa dirumuskan sebagai berikut: kita ingin menemukan set parameter (w, b) sehingga sgn(f(xi)) =sgn(< w,x > +b) = yi untuk semua i. Fungsi sgn digunakan untuk mengelompokkan semua nilai di atas 0 menjadi +1 dan dibawah 0 menjadi -1. Dalam teknik ini kita berusaha menemukan fungsi pemisah (klasifier/hyperplane) terbaik diantara fungsi yang tidak terbatas jumlahnya untuk memisahkan dua macam obyek. Hyperplane terbaik adalah hyperplane yang terletak di tengah-tengah antara dua set obyek dari dua kelas. Mencari hyperplane terbaik ekuivalen dengan memaksimalkan margin atau jarak antara dua set obyek dari kelas yang berbeda. Jika wx1 + b = +1 adalah hyperplane-pendukung (supporting hyperplane) dari kelas +1, dan wx2 + b = -1 adalah hyperplanependukung dari kelas -1, margin antara dua kelas dapat dihitung dengan mencari jarak antara kedua hyperplane-pendukung dari kedua kelas. Secara spesifik, margin dihitung dengan cara berikut :

wx1 + b = +1

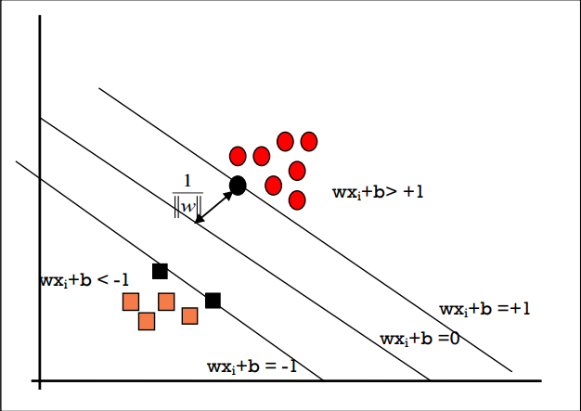
wx2 + b = -1

------------------- -

w(x1 - x2) = 2

→

Gambar tersebut memperlihatkan bagaimana SVM bekerja untuk menemukan suatu fungsi pemisah dengan margin yang maksimal. Untuk membuktikan bahwa memaksimalkan margin antara dua set obyek akan meningkatkan probabilitas pengelompokkan secara benar dari data testing. Pada dasarnya jumlah fungsi pemisah ini tidak terbatas banyaknya. Misalkan dari jumlah yang tidak terbatas ini kita ambil dua saja, yaitu f1(x) and f2(x) (lihat gambar 8.2). Fungsi f1 mempunyai margin yang lebih besar dari pada fungsi f2. Setelah menemukan dua fungsi ini, suatu data baru masuk dengan keluaran −1. Kita harus mengelompokkan apakah data ini ada dalam kelas −1 atau +1 menggunakan fungsi pemisah yang sudah kita temukan.



Gambar 5 Metode SVM

Sumber : <https://medium.com/@samsudiney/penjelasan-sederhana-tentang-apa-itu-svm-149fec72bd02>

Mencari fungsi pemisah yang optimal untuk obyek yang bisa dipisahkan secara linier Dengan menggunakan f1, kita akan kelompokkan data baru ini di kelas −1 yang berarti kita benar mengelompokkannya. Sekarang kita coba gunakan f2, kita akan menempatkannya di kelas +1 yang berarti salah. Dari contoh sederhana ini kita lihat bahwa memperbesar margin bisa meningkatkan probabilitas pengelompokkan suatu data secara benar. [ ]

Adapun tahap *modelling* pada fase ini adalah sebagai berikut.

|  |  |
| --- | --- |
| [17] |  |
| [18] |  |
| [19] |  |
| [20] |  |
| [21] |  |
| [22] |  |
| [23] |  |
| [24] |  |
| [25] |  |
| [26] |  |

Keterangan :

1. Baris 17 : Menyimpan dataframe menjadi dummy pada atribut “diagnosis“ pada variabel “Diagnosis“. Ketika ditampilkan 5 data teratas, atribut berubah menjadi “1“
2. Baris 18 : Merubah atribut dataframe “Diagnosis“ yang awalnya “1“ menjadi “diagnosis“
3. Baris 19 : Mengubah seluruh nilai dataframe “Diagnosis“ dimana nilai “1“ menjadi “Tumor Ganas (Terdeteksi Kanker)“ dan nilai “0“ menjadi “Tumor Jinak (Tidak Terdeteksi Kanker)“
4. Baris 20 : Menghapus atribut “diagnosis“ pada df
5. Baris 21 : Menggabungkan dataframe “df“ dan dataframe “Diagnosis“
6. Baris 22 : Menyimpan dataframe “Diagnosis“ pada variabel “y“
7. Baris 23 : Menghapus atribut “diagnosis“ pada df, kemudian disimpan dalam variabel “X“
8. Baris 24 : Melakukan split data menjadi data training (sebanyak 80%) dan data testing (sebanyak 20%) menggunakan library train\_test\_split
9. Baris 25 : Membuat model menggunakan Support Vector Machine  dan membuat ukuran kesesuaian model menggunakan library SVC. Kemudian model disimpan kedalam variabel “Persamaan“
10. Baris 26 : Melakukan cek hasil prediksi menggunakan dataset testing X\_test

## *Evaluation*

Tahap *evaluation* lebih berfokus untuk melihat apakah pemodelan yang dilakukan memiliki hasil yang baik atau malah sebaliknya. Pada tahap ini, evaluasi model dilakukan dengan menggunakan metode akurasi dan *confussion* *matrix* untuk melihat hasil-hasil evaluasi dari model.

Adapun tahap *evaluation* pada fase ini adalah sebagai berikut.

|  |  |
| --- | --- |
| [26] |  |
| [27] |  |
| [28] |  |
| [29] |  |

Keterangan :

1. Baris 26 : Membuat dan menampilkan confusion matrix untuk mengetahui akurasi model.
2. Baris 27 : Membuat dan menampilkan confusion matrix untuk mengetahui akurasi model menggunakan seaborn agar lebih mudah dibaca.
3. Baris 28 : Menampilkan hasil akurasi model prediksi yang dibuat menggunakan library accuracy\_score
4. Baris 29 : Melakukan percobaan prediksi dengan inputan manual.

## *Deployment*

Yang terakhir ialah tahap *deployment*. Dimana pada tahap ini, hasil model penelitian yang telah dilakukan akan dikembangkan dan didokumentasikan. Pada tahap ini, hal yang dilakukan dalam fase ini yaitu mengembangkan model ke dalam bentuk aplikasi sehingga lebih mudah digunakan dalam melakukan proses prediksi.

Adapun tahap *deployment* pada fase ini adalah sebagai berikut.

|  |  |
| --- | --- |
| [30] |  |
| [31] |  |

Keterangan :

1. Baris 30 : Mengimport library pickle untuk menyimpan data
2. Baris 31 : Menyimpan model (variabel “persamaan“) kedalam bentuk pickle, dan memberi nama file menjadi “model.pickle“

# BAB 3 Apa Saja Perangkat Yang Digunakan Untuk Membuat Aplikasi?

## Bahasa Pemrograman Yang Digunakan

### **Python**



Gambar 10 Logo Python

Sumber : <https://www.python.org/community/logos/>

Python adalah bahasa pemrograman interpretatif multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode. Pyhton disebutkan sebagai bahasa yang menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang sangat jelas, dan dilengkapi dengan fungsionalitas pustaka standat yang besar serta kompreherensif. Python mendukung multi paradigma pemrograman, utamanya; namun tidak dibatasi; pada pemrograman berorientasi objek, pemrograman imperatif, dan pemrograman fungsional. Salah satu fitur yang tersedia pada python adalah sebagai bahasa pemrogaman dinamis yang dilengkapi dengan manajamen memori otomatis.

Seperti halnya bahasa pemrograman dinamis lainnya, python umumnya digunakan sebagai bahasa *script* meski pada praktiknya penggunaan bahasa ini lebih luas mencakup konteks pemanfaatan yang umumnya tidak dilakukan dengan menggunakan bahasa *script*. Python dapat digunakan untuk berbagai keperluan pengembangan perangkat lunak dan dapat berjalan di berbagai platform sistem operasi. Saat ini kode python dapat dijalankan di berbagai platform sistem operasi, beberapa diantaranya adalah :

1. Linux/Unix
2. Windows
3. Max OS X
4. Java Virtual Machine
5. Amiga
6. Palm
7. Symbian (untuk produk-produk Nokia)

Python didistribusikan dengan beberapa lisensi yang berbeda dari beberapa versi. Namun pada prinsipnya Python dapat diperoleh dan dipergunakan secara bebas, bahkan untuk kepentingan komersial. Lisensi Python tidak bertentangan baik menurut definisi Open Source maupun General Public License (GPL).

Bahasa pemrograman Python tentunya tidak dapat diemplementasikan atau digunakan secara langsung pada *software* teks editor ataupun terminal pada *device* seperti bahasa pemrograman PHP. Bahasa pemrograman ini akan dapat digunakan jika telah ter-install pada *device* yang digunakan.

### **HTML**



Gambar 11 Logo HTML

Sumber : <https://www.kindpng.com/imgv/iJJbTR_java-html-language-logo-png-transparent-png/>

HTML (*Hyper Text Mark Up Language*) ialah bahasa yang dipergunakan dengan tujuan untuk mendeskripsikan struktur dar sebuah halaman web. HTML memiliki fungsi untuk mempublikasikan dokumen secara online. *Tags* adalah statement dasar dari HTML. Sebuah *tag* dinyatakan dalam kurung siku (<>). *Tags* yang ditujukan untuk sebuah dokumen atau bagian dari suatu dokumen harus dibuat dalam bentuk berpasangan. *Tag* terdiri dari dua macam yaitu *tag* pembuka dan *tag* penutup. Dimana *tag* penutup menggunakan tambahan tanda garis miring (/) di awal nama *tag*(Pahlevi et al., 2018).

### **CSS**



Gambar 12 Logo CSS

Sumber :

<https://commons.wikimedia.org/wiki/File:CSS3_logo_and_wordmark.svg>

CSS memiliki kepanjangan *Cascading Style Sheet* adalah bahasa-bahasa yang dapat merepresentasikan halaman web. Representasi tersebut diantaranya adalah warna, *layout*, dan *font*. Dengan digunakannya CSS dalam membangun halaman web, seorang web developer dapat membuat halaman web yang dapat beradaptasi dengan berbagai macam ukuran layar. Pembuatan skrip CSS biasanya terpisah dengan halaman HTML. Meskipun skrip CSS dapat disisipkan di dalam halaman HTML. Hal ini ditujukan untuk memudahkan pengaturan halaman HTML yang memiliki rancangan yang sama (Henderson, 2009:72) (Pahlevi et al., 2018).

### **Bootstrap**



Gambar 13 Logo Bootstrap

Sumber : <https://pkp.sfu.ca/2016/09/02/bootstrap-theme-for-ojs-3/>

Bootstrap adalah suatu *framework* untuk CSS (*Cascading Style Sheet*) dan berupa produk bentuk opensource (terbuka untuk siapapun) yang dibuat oleh Mark Otto dan Jacob Thornton. Pada awalnya Bootstrap ini dibuat untuk membuat standarisasi dalam front end untuk seluruh programmer yang ada di perusahaannya. Bootsrap telah berubah dari yang sebelumnya adalah CSS-Driven proyek ke sebuah host dari JavaScript plugins dan ikon yang dapat dengan mudah digunakan untuk *form*ulir dan tombol.

### **Django**



Gambar 14 Logo Django

Sumber : <https://www.djangoproject.com/community/logos/>

Django ialah sebuah web *framework* berbasis bahasa pemrograman Python yang didesain untuk membuat suatu aplikasi web yang dinamis, kaya fitur dan aman. (Saputra & Fathoni Aji, 2018) Django yang dikembangkan oleh Django *Software* Foundation terus mendapatkan perbaikan sehingga membuat web *framework* yang satu ini menjadi pilihan utama bagi banyak pengembang aplikasi web. (Saputra & Fathoni Aji, 2018)

*Framework* Django bertujuan untuk memudahkan pengembangan situs web dan basis data yang kompleks (Django *Software* Foundation, 2017). Salah satu keunggulan django yaitu *framework* ini telah merepresentasikan model ORM (*Object Relational Mapper*) sehingga tidak perlu lagi untuk menyesuaikan *query* jika terjadi perubahan *database* yang digunakan. Berikut ini adalah beberapa keuntungan menggunakan Django:

1. *Object-Relational Mapping (ORM) Support.*
2. *Automatic Admin Interface.*
3. *Elegant URL Design.*
4. *Template System.*
5. *Cache system.*
6. *Internationalization.*
7. *A light weight web server for development and test*(Rahimah et al., 2019).

## Penyimpanan Yang Digunakan

### **Database MySQL**



Gambar 15 Logo MySQL

Sumber : <https://www.logo.wine/logo/MySQL>

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (Bahasa Inggris: *database management system*) atau DBMS yang *multihead*, *multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL adalah DBMS yang *open source* dengan dua bentuk lisensi, yaitu *Free Software* (perangkat lunak bebas) dan *Shareware* (perangkat lunak berpemilik yang penggunaannya terbatas). Jadi MySQL adalah *database* *server* yang gratis dengan lisensi GNU *General Public License* (GPL) sehingga dapat dipakai untuk keperluan pribadi atau komersil tanpa harus membayar lisensi yang ada. Istilah semacam baris, kolom, tabel, dipakai pada MySQL. SQL sendiri merupakan suatu Bahasa yang dipakai di dalam pengambilan data pada *relational database* atau *database* yang terstruktur. Jadi MySQL adalah *database management system* yang menggunakan Bahasa SQL sebagai Bahasa penghubung antara perangkat lunak aplikasi dengan *database server*. MySQL sendiri sebenarnya merupakan lanjutan dari proyek UNIREG yang dikerjakan oleh Michael Montry Widenius dan TcX (perusahaan perangkat lunak asal Swedia). Sayangnya, UNIREG belum terlalu kompatibel dengan *database* dinamis yang dipakai di *website*. TcX kemudian mencari alternatif lain dan menemukan perangkat lunak yang dikembangkan oleh David Hughes, yaitu miniSQL atau mSQL. Namun, ditemukan masalah lagi karena mSQL tidak mendukung *indecing* sehingga belum sesuai dengan kebutuhan TcX. MySQL *server* mendukung beberapa macam sistem operasi (*cross-platform*), khususnya Linux dan Windows. Proses instalasinya pun sangat mudah. Jika menggunakan Windows, bisa menginstall XAMPP untuk menjalankan MySQL *server* yang di dalamnya sudah terdapat juga modul untuk menjalankan Apache, PHP, FileZilla, dan Tomcat. Sedangkan di Linux, dapat menginstall MySQL secara terpisah atau menginstall LAMp (Linux, Apache, MySQL, PHP) yang sudah ada pada modul Apache dan PHP. Selain itu, XAMPP juga sudah tersedia di Linux(Hasan Bisri Isa Alfaris et al., 2013).

## Perangkat Lunak Yang Dibutuhkan

### **Visual Studio Code**



Gambar 16 Visual Studio Code

Sumber : <https://id.wikipedia.org/wiki/Visual_Studio_Code>

Visual Studio Code (VS Code) adalah sebuah teks editor ringan dan handal yang dibuat oleh Microsoft untuk sistem operasi *multiplatform*, artinya tersedia juga untuk versi Linux, Mac, dan Windows. Teks editor ini secara langsung mendukung bahasa pemrograman dengan bantuan plugin yang dapat dipasang via *marketplace* Visual Studio Code(Yudo Ardianto Agung, 2020).

Berikut fitur-fitur unggulan Visual Studio Code :

1. *Cross platform*

Dapat berjalan di Windows, Linux, dan MacOS. Artinya dapat bekerja pada sistem operasi manapun dengan menggunakan Visual Studio Code.

1. *Lightweight*

Tak perlu menunggu lama untuk memulai. Bahasa, tema, debugger, commands, dan lain-lainnya dapat dikontrol sepenuhnya sesuai keinginan.

1. *Powerfull Editor*

*Source code editing* yang sangat produktif, seperti membuat code *snippets*, *IntelliSense*, *auto correct*, dan format*ting*.

1. *Code Debugging*

Salah satu fitur bagus yang ditawarkan Visual Studio Code adalah membantu untuk melakukan debug pada kode dengan cara mengawasi kode, variabel, *call stack* dan *expression* yang mana saja.

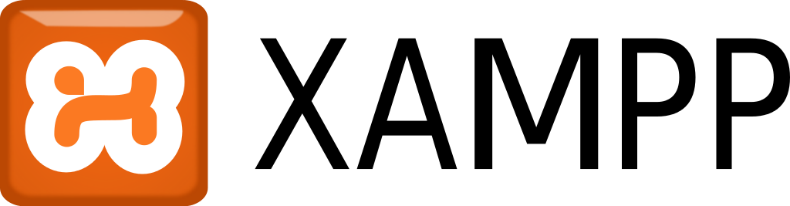
1. *Source Control*

Visual Studio Code memiliki *intgrated source control* termasuk *Git support in-the-box* dan penyedia *source code control* lainnya di pasaran. Ini meningkatkan siklus rilis proyek secara signifikan.

1. *Intergrated* *Terminal*

Tidak ada lagi *multiple* Windows dan Alt-Tabs. Perintah tersebut dapat dilakukan dengan *command-line task* sekejap dan membuat banyak terminal di dalam editor.

### **XAMPP**



Gambar 17 Logo Xampp

Sumber [: https://id.m.wikipedia.org/wiki/Berkas:Xampp\_logo.svg](:%20https:/id.m.wikipedia.org/wiki/Berkas:Xampp_logo.svg)

XAMPP adalah sebuah perangkat lunak komputer yang sistem penamaannya diambil dari akronim kata Apache, MySQL/MariaDB, PHP, dan Perl. Sementara imbuhan huruf “X“ yang terdapat pada awal kata berasal dari istilah *cross platform* sebagai simbol bahwa aplikasi ini bisa dijalankan di empat sistem operasi berbeda, seperti Linux, Windows, MacOS, dan juga Solaris. Jika dijabarkan, masing-masing huruf yang ada di dalam nama XAMPP menurut para ahli memiliki arti sebagai berikut ini:

1. X (*Cross Platform*)

Merupakan kode penanda untuk *software cross platform* atau yang bisa berjalan di banyak sistem operasi. Jadi, ada XAMPP untuk Windows, MacOS, Linux. Semua itu bersifat *free* download XAMPP.

1. A (*Apache*)

Apache adalah aplikasi *web server* yang bersifat gratis dan bisa dikembangkan oleh banyak orang (*open source*).

1. M (*MySQL/MariaDB*)

MySQL atau MariaDB merupakan aplikasi *database server* yang dikembangkan oleh orang yang sama. MySQL berperan dalam mengolah, mengedit, dan menghapus daftar melalui *database*.

1. P (*PHP*)

Huruf “P“ yang pertama dari akronim kata XAMPP adalah inisial untuk menunjukkan eksistensi bahasa pemrograman PHP. Bahasa pemrograman ini biasanya digunakan untuk membuat *website* dinamis, contohnya dalam *website* berbasis CMS WordPress.

1. P (*Perl*)

Sementara itu, huruf P selanjutnya merupakan singkatan dari bahasa pemrogaman Perl yang kerap digunakan untuk memenuhi berbagai macam kebutuhan. Perl ini bisa berjalan di dalam banyak digunakan.

Program aplikasi XAMPP berfungsi sebagai *server* lokal untuk mengampu berbagai jenis data *website* yang sedang dalam proses pengembangan. Dalam prakteknya, XAMPP digunakan untuk menguji kinerja fitur ataupun menampilkan konten yang ada di dalam *website* kepada orang lain tanpa harus terkoneksi dengan internet, cukup akses melalui XAMPP *Control panel*, atau istilahnya *website offline*. XAMPP bekerja secara *offline* layaknya *web hosting* biasa namun tidak bisa diakses oleh banyak orang.

### **Jupyter Notebook**



Gambar 18 Logo Jupyter

Sumber : <https://icon-icons.com/id/icon/jupyter-logo/169453>

Jupyter Notebook adalah lingkungan komputasi interaktif berbasis *web* untuk membuat dokumen *notebook* jupyter. Jupyter notebook mendukung beberapa bahasa seperti Python (Ipython), Julia, R, dan lain-lain. Sebagian besar digunakan untuk analisis data, visualisasi data dan komputasi eksplorasi interaktif lebih lanjut. Awalnya dikembangkan untuk aplikasi *data science* yang ditulis dengan Python, R, dan Julia, Jupyter Notebook berguna dalam segala macam cara untuk semua jenis proyek :

1. Visualisasi Data

Kebanyakan orang memiliki eksposur pertama mereka ke Notebook Jupyter melalui visualisasi data, sebuah *notebook* bersama yang mencakup *rendering* dari beberapa data yang ditetapkan sebagai grafik. Jupyter Notebook memungkinkan untuk membuat visualisasi, tetapi juga membagikannya dan memungkinkan perubahan interaktif pada kode dan kumpulan data yang dibagikan.

1. Berbagi Kode

Layanan *cloud* seperti GitHub dan Pastebin menyediakan cara untuk berbagi kode, tetapi sebagian besar non-interaktif. Dengan Jupyter Notebook dapat dilihat kode, menjalankannya, dan menampilkan hasilnya langsung di *browser web*.

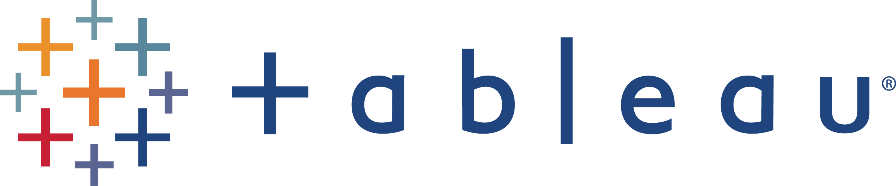
1. Interaksi Langsung dengan Kode

Kode Notebook Jupyter tidak statis, itu dapat diedit dan dijalankan kembali secara bertahap, dengan umpan balik yang diberikan langsung di *browser*. Notebook juga dapat menyematkan kontrol pengguna (Misalnya Slider atau bidang input teks) yang dapat digunakan sebagai sumber input untuk kode.

1. Mendokumentasikan Contoh Kode

Jika memiliki sepotong kode dan ingin menjelaskan secara bertahap cara kerjanya, dengan umpan balik langsung di sepanjang jalan, dapat menyematkannya dalam Jupyter Notebook. Kode akan tetap berfungsi penuh, dapat menambahkan interaktivitas bersama dengan penjelasan, yang ditampilkan dan diceritakan pada saat yang sama(small-business-tracker.com, 2022).

### **Tableau**



Gambar 19 Logo Tableau

Sumber : <https://logos-world.net/tableau-logo/>

Tableau adalah *platform business intelligence* untuk membantu dalam melihat dan memahami data. *Platform* ini merupakan hasil dari proyek ilmu komputer di Stanford yang didirikan pada tahun 2003. Tujuan dibuatnya Tableau adalah untuk meningkatkan aliran analisis dan membuat data lebih dapat diakses oleh orang-orang melalui visualisasi. Berikut keunggulan menggunakan Tableau :

1. Visualisasi yang mudah dan interaktif

Keunggulan yang membuat populer Tableau adalah visualisasinya mudah, interaktif, dan menarik. Jika *tools* lain hanya menyajikan grafik biasa, *tools* ini bisa menyajikan grafik bergerak, sehingga tampilan lebih menarik.

1. *User-friendly*

Jika membuat grafik perlu adanya *coding*, Tableau tidak perlu ada *coding*. Tableau diperuntukan bagi siapapun yang bermain dengan data tapi ingin lebih sederhana dalam menyajikannya.

1. Banyak sumber data yang mampu diolah

Banyak data kompleks yang beraturan maupun tidak beraturan, dan juga berbagi sumber, Tableau bisa mengolah dari berbagai sumber. Bahkan bisa menggabungkan dari berbagai *cloud*, *spreadsheet*, dan *big data*.

1. *Dashboard mobile friendly*

Jika tidak sempat membuka laptop, Tableau bisa diakses melalui *smarthphone*. Meski lewat smartphone, *user interface*nya mudah untuk dioperasikan.

1. Terintegrasi dengan bahasa srkip(Institut Teknologi Bandung, 2022).

### **Anaconda**



Gambar 20 Logo Anaconda

Sumber : <https://en.wikipedia.org/wiki/Anaconda_%28Python_distribution%29>

Anaconda merupakan aplikasi gratis yang memiliki banyak package dan tools atau dengan kata lain merupakan *one place solution platform* untuk *data scientist* agar memudahkan dalam mengerjakan proyek karena di dalamnya terdapat berbagai macam *software*. Anaconda bisa dikatakan sebagai paket distribusi python dari *continuum analytics* yang berisi paket *Python* ditambah beberapa paket tambahan untuk keperluan pemrograman *data science*, matematika, hingga teknik dalam satu distribusi yang *user friendly*.

### **Kaggle**



Gambar 21 Logo Kaggle

Sumber : <https://commons.wikimedia.org/wiki/File:Kaggle_logo.png>

Kaggle adalah situs dan platform untuk berlomba membuat model terbaik untuk menganalisa dan memprediksi suatu dataset. Dataset ini diberikan oleh suatu perusahaan, dengan suatu deskripsi masalah tertentu. Misalnya, diberikan data rumah beserta fitur-fiturnya dan harga jualnya, dan deskripsi masalahnya adalah buatlah model untuk memprediksi harga jual berdasarkan fitur-fitur tersebut.

# BAB 4 Perancangan Sistem UML dalam Membuat Aplikasi

## Analisis

Analisis sistem adalah pembagian suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya yang bertujuan untuk mengevaluasi masalah atau kendala yang terjadi pada suatu sistem, sehingga nantinya dapat dilakukan perbaikan atau pengembangan pada sistem tersebut.

Perancangan sistem adalah kegiatan merancang dan mendesain suatu sistem yang baik, yang dimana kegiatan tersebut adalah langkah-langkah operasi dalam proses pengolahan data dan prosedur-prosedur untuk mendukung operasi sistem tersebut.

Tujuan dari perancangan sistem adalah untuk memenuhi kebutuhan para pengguna sistem serta memberikan gambaran yang jelas dan rancang bangun yang lengkap kepada programmer dan ahli-ahli yang terlibat didalamnya. Pada bagian ini, dibahas tentang analisis prosedur yang digambarkan dalam bentuk flowmap BPMN, pengkodean, analisis sistem fungsional, dan analisis sistem non fungsional yang meliputi perangkat keras dan perangkat lunak yang digunakan. Selain itu, pada bagian ini juga akan dibahas mengenai analisis pengguna yang terlibat dalam aplikasi tersebut. Tahapan ini sangat penting dalam membantu melanjutkan tahapan selanjutnya yaitu tahapan perancangan.

Top of Form

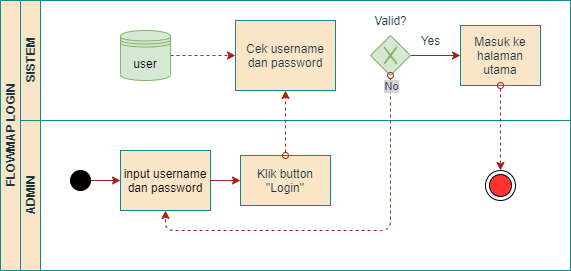
Bottom of Form

### **Analisis Sistem Yang Akan di Bangun**

1. ***Flowmap* *Login***

Pada bagian ini, akan dijelaskan tentang aplikasi website yang akan dibangun untuk memprediksi kanker payudara pada pasien berdasarkan faktor-faktor spesifik. Analisis sistem yang sedang berjalan bertujuan untuk memberikan gambaran dan mengetahui lebih lanjut bagaimana cara kerja sistem tersebut. Strategi yang digunakan dalam menganalisis sistem yang akan dibangun ini adalah dengan memecah atau menterjemahkan ke dalam bentuk flowmap BPMN.

Berikut adalah *flowmap* *login* yang akan dibangun dalam pembuatan aplikasi prediksi gaji.



Keterangan :

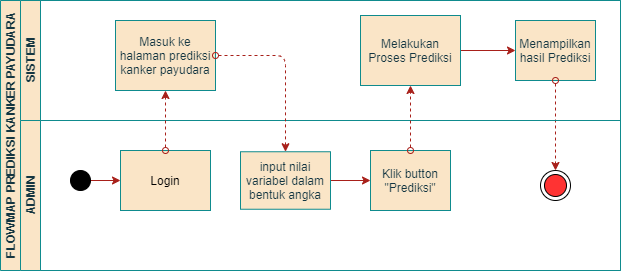
1. Admin dapat membuka Aplikasi Prediksi Kanker Payudara
2. Aplikasi menampilkan halaman *login* .
3. Admin dapat melakukan penginputan *username* serta *password*.
4. Setelah itu, sistem dapat melakukan pengecekan apakah *username* serta *password* yang diinputkan admin valid atau tidak.
5. Jika *username* dan *password* sesuai, maka admin dapat masuk ke halaman .utama aplikasi. Sedangkan jika *username* dan *password* tidak sesuai, maka admin akan tetap berada di halaman *login*.
6. ***Flowmap* Prediksi Kanker Payudara**

Pada bagian ini menjelaskan tentang *flowmap* proses prediksi kanker payudara yang dilakukan oleh admin. *Flowmap* tersebut merupakan gambaran alur proses prediksi kanker payudara oleh admin yang akan dibangun pada aplikasi ini.

Pada *flowmap* ini, terdapat dua pelaku dalam kegiatan, diantaranya adalah:

1. Admin sebagai pengguna sistem.
2. Sistem sebagai komponen informasi yang akan digunakan.

Berikut adalah *flowmap* proses prediksi kanker payudara yang akan dibangun dalam pembuatan pada aplikasi.



Keterangan :

1. Admin dapat melakukan *login*.
2. Setelah admin berhasil *login*, aplikasi akan menampilkan halaman *dashboard*.
3. Setelah itu, admin dapat memilih menu Prediksi untuk beralih ke halaman prediksi.
4. Pada halaman prediksi, admin dapat menginputkan angka berupa variabel *form* yang disediakan.
5. Setelah diinputkan, aplikasi akan menampilkan hasil prediksi kanker payudara pasien.

## Perancangan Sistem ( UML )

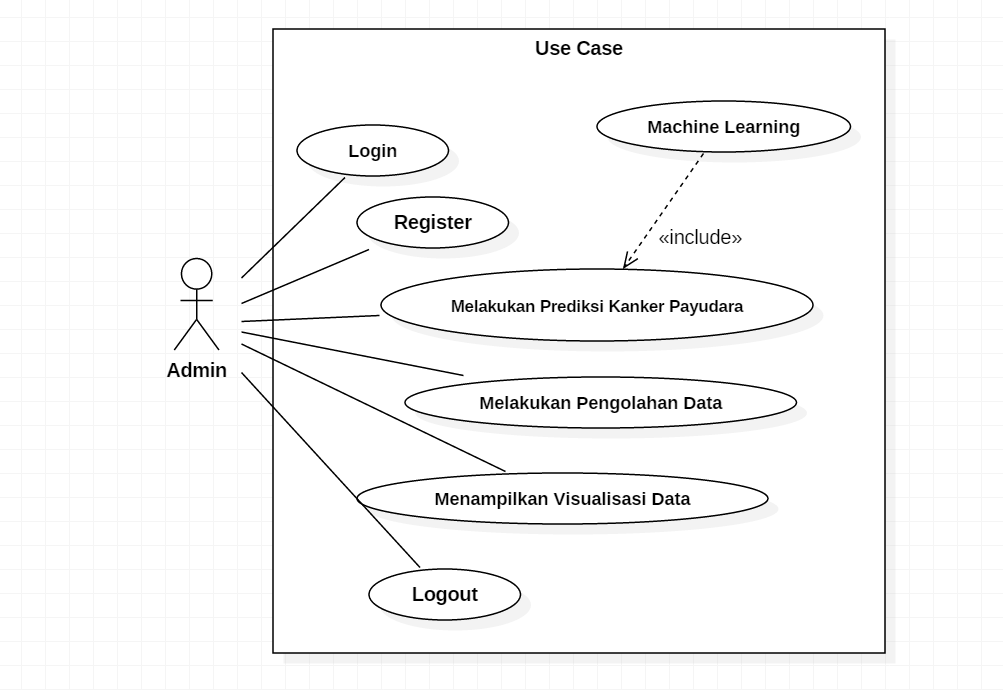
UML adalah sebuah bahasa yang digunakan untuk memodelkan sistem atau aplikasi berbasis objek. Ia merupakan standar spesifikasi yang digunakan untuk dokumentasi, spesifikasi, dan pembangunan sistem perangkat lunak. UML juga merupakan metodologi untuk mengembangkan sistem OOP (Object-Oriented Programming) dan memiliki alat-alat yang membantu dalam pengembangan sistem tersebut. Dengan demikian, UML dapat menjadi cara yang memudahkan dalam pengembangan aplikasi yang berkelanjutan dan bermanfaat sebagai alat bantu untuk mentransfer pengetahuan tentang sistem yang akan dikembangkan dari satu developer ke developer lainnya.

### **Use case Diagram**

Use case diagram adalah salah satu jenis diagram UML (Unified Modeling Language) yang digunakan untuk menggambarkan interaksi sistem dengan aktor-aktor yang terlibat dalam sistem tersebut. Diagram ini memfokuskan pada fungsi sistem dan cara aktor terlibat dengan sistem tersebut.

Use case diagram terdiri dari beberapa elemen utama, yaitu:

1. Aktor: merupakan entitas yang terlibat dalam interaksi dengan sistem, baik itu manusia maupun perangkat lain. Aktor dapat diwakili oleh manusia, organisasi, atau perangkat lain yang menggunakan sistem.
2. Use case: merupakan tindakan atau kegiatan yang dapat dilakukan oleh sistem dengan bantuan aktor. Use case menggambarkan bagaimana sistem memenuhi kebutuhan aktor.
3. Association: merupakan garis yang menghubungkan aktor dengan use case. Association menggambarkan bagaimana aktor terlibat dalam use case.
4. Generalization: merupakan garis yang menghubungkan use case dengan use case lainnya. Generalization menggambarkan bahwa use case yang terhubung memiliki kesamaan atau kemiripan dalam tindakannya.



1. Definisi Aktor

Pada bagian ini akan dijelaskan aktor-aktor yang terlibat dalam Sistem.

|  |  |  |
| --- | --- | --- |
| No | Aktor | Deskripsi |
| 1 | Admin | * *Login* * Melakukan Prediksi Kanker Payudara * Melakukan Pengolahan Data * Melakukan Visualisasi Data * *Logout* |

1. Definisi *Use case*

|  |  |  |
| --- | --- | --- |
| No | Aktor | Deskripsi |
| 1 | *Machine learning* | Merupakan aktivitas interaksi antara manusia dengan mesin. Dalam hal ini, admin membuat model prediksi *Machine learning*. |
| 2 | *Login* | Merupakan aktivitas *login* yang dilakukan oleh admin. Sebelum admin masuk ke aplikasi ia harus melakukan *login* terlebih dahulu. |
| 3 | Registrasi | Merupakan aktivias registrasi yang dilakukan oleh admin. Apabila admin belum memiliki akun, maka admin harus melakukan registrasi terlebih dahulu. |
| 4 | Melakukan Prediksi Gaji | Merupakan aktivitas memprediksi kanker pada pasien yang dilakukan oleh admin dengan parameter yang telah diperiksa oleh dokter. |
| 5 | Melakukan Pengolahan Data | Merupakan aktivitas *insert*, *read*, *update* dan *delete* data pasien yang dilakukan oleh admin. |
| 6 | Menampilkan Visualisasi Data | Merupakan aktivitas visualisasi grafik data pasien yang dilakukan oleh admin. |
| 7 | *Logout* | Merupakan aktivitas *Logout* yang dilakukan oleh admin. Admin dapat keluar dari aplikasi jika telah selesai melakukan pekerjaan pada aplikasi. |

1. Skenario *Use case*

Skenario *use case* diharapkan setelah berjalannya fungsional *use case*. Selain itu juga diberikan ulasan yang berkaitan dengan tanggapan dari sistem atas suatu aksi yang dilakukan oleh aktor. Setiap *use case* akan diberikan sebuah skenario yang akan menjelaskan secara detail interaksi yang ada di dalamnya.

Skenario Use case Machine learning

|  |  |
| --- | --- |
| Identifikasi | |
| Nomor | 1 |
| Nama | *Machine learning* |
| Tujuan | Pembuatan model prediksi |
| Deskripsi | |
| Aktor | Admin |
| Skenario Utama | |
| Kondisi Awal |  |
| Aksi Aktor | Reaksi Sistem |
| Membuat model prediksi | Melakukan olah data pasien |
| Kondisi Akhir | Masuk pada aplikasi |

Skenario Use case Login Admin

|  |  |
| --- | --- |
| Identifikasi | |
| Nomor | 1 |
| Nama | *Login* |
| Tujuan | Sebelum masuk pada aplikasi |
| Deskripsi | |
| Aktor | Admin |
| Skenario Utama | |
| Kondisi Awal |  |
| Aksi Aktor | Reaksi Sistem |
| Input *username* dan *password* | Mengecek data dan menentukan aktor tersebut admin atau *user* |
| Kondisi Akhir | Masuk pada aplikasi |

Skenario Use case Registrasi

|  |  |
| --- | --- |
| Identifikasi | |
| Nomor | 1 |
| Nama | Registrasi |
| Tujuan | Membuat akun admin |
| Deskripsi | |
| Aktor | Admin |
| Skenario Utama | |
| Kondisi Awal |  |
| Aksi Aktor | Reaksi Sistem |
| Input *username*, dan *password* | Menyimpan data admin atau *user* |
| Kondisi Akhir | Masuk pada halaman *login* |

Skenario Use case Melakukan Prediksi

|  |  |
| --- | --- |
| Identifikasi | |
| Nomor | 1 |
| Nama | Melakukan Prediksi Kanker Payudara |
| Tujuan | Melakukan Prediksi Kanker Payudara |
| Deskripsi | |
| Aktor | Admin |
| Skenario Utama | |
| Kondisi Awal |  |
| Aksi Aktor | Reaksi Sistem |
| Menginput Data | Melakukan Prediksi Kanker Payudara |
| Kondisi Akhir | Admin dapat memprediksi Kanker Payudara |

Skenario Use case Melakukan Visualisasi Data

|  |  |
| --- | --- |
| Identifikasi | |
| Nomor | 1 |
| Nama | Melakukan Visualisasi Data |
| Tujuan | Melakukan visualisasi data Pasien |
| Deskripsi | |
| Aktor | Admin |
| Skenario Utama | |
| Kondisi Awal |  |
| Aksi Aktor | Reaksi Sistem |
| Input data karyawan | Menampilkan grafik visualisasi data pasien |
| Kondisi Akhir | Admin dapat memvisualisasikan data |

Skenario Use case Melakukan Pengolahan Data

|  |  |
| --- | --- |
| Identifikasi | |
| Nomor | 1 |
| Nama | Melakukan Pengolahan Data |
| Tujuan | Melakukan pengolahan data Pasien |
| Deskripsi | |
| Aktor | Admin |
| Skenario Utama | |
| Kondisi Awal |  |
| Aksi Aktor | Reaksi Sistem |
| *Insert*, *read*, *update*, dan *delete* data karyawan. | Mengubah data Pasien yang diubah oleh admin |
| Kondisi Akhir | Admin dapat mengelola data |

Skenario Use case Logout Admin

|  |  |
| --- | --- |
| Identifikasi | |
| Nomor | 1 |
| Nama | *Logout* |
| Tujuan | Keluar dari aplikasi |
| Deskripsi | |
| Aktor | Admin |
| Skenario Utama | |
| Kondisi Awal |  |
| Aksi Aktor | Reaksi Sistem |
| Menekan tombol *Logout* | Melakukan proses keluar dari aplikasi |
| Kondisi Akhir | Keluar dari aplikasi |

* + 1. **Class Diagram**

Class diagram adalah salah satu jenis diagram UML (Unified Modeling Language) yang digunakan untuk menggambarkan struktur dan relasi antar kelas dalam suatu sistem. Diagram ini menggambarkan bagaimana kelas-kelas tersebut saling berinteraksi dan berkaitan satu sama lain dalam sistem yang dibuat.

Class diagram terdiri dari beberapa elemen utama, yaitu:

1. Class: merupakan entitas yang mewakili suatu objek dalam sistem. Class menggambarkan atribut (properti) dan operasi (method) yang dimiliki oleh objek tersebut.
2. Attribute: merupakan properti yang dimiliki oleh sebuah kelas. Attribute menggambarkan data yang disimpan oleh kelas tersebut.
3. Operation: merupakan method yang dimiliki oleh sebuah kelas. Operation menggambarkan tindakan atau kegiatan yang dapat dilakukan oleh kelas tersebut.
4. Association: merupakan garis yang menghubungkan satu kelas dengan kelas lainnya. Association menggambarkan relasi atau hubungan antar kelas.
5. Generalization: merupakan garis yang menghubungkan kelas dengan superclass-nya. Generalization menggambarkan bahwa kelas tersebut merupakan turunan dari superclass yang terhubung.
   * 1. **Activity Diagram**

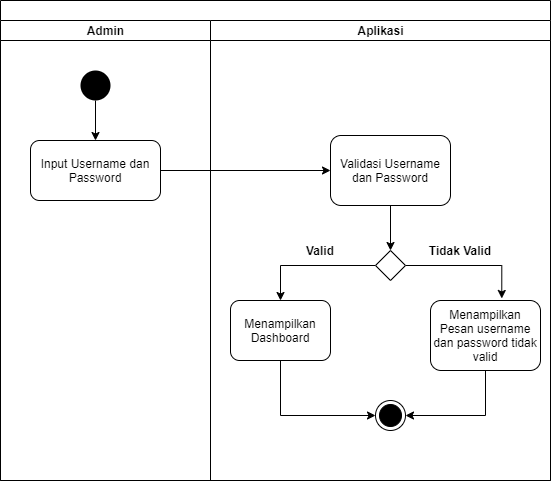
Activity diagram adalah salah satu jenis diagram UML (Unified Modeling Language) yang digunakan untuk menggambarkan aliran aktivitas dalam suatu sistem. Diagram ini menggambarkan langkah-langkah yang dilakukan dalam suatu proses atau kegiatan, serta kondisi yang dapat mempengaruhi aliran aktivitas tersebut.

1. *Activity diagram* *Login* Admin

Berikut ini merupakan *Activity diagram* *Login* Admin yang menggambarkan tentang berbagai aktifitas yang terjadi pada suatu sistem. Dimana pada diagram ini, terdiri dari komponen-komponen sebagai berikut.

1. Aktor : Admin dan Aplikasi
2. Satu *initial node*
3. *Activity* : Terdapat empat *activity* yang dapat dipaparkan sebagai berikut.
4. Input *username* dan *password*
5. Validasi *username* dan *password*
6. Menampilkan *dashboard*
7. Menampilkan pesan *username* dan *password* tidak valid
8. Satu percabangan untuk valid dan tidak valid
9. Satu *Activity Final Node*

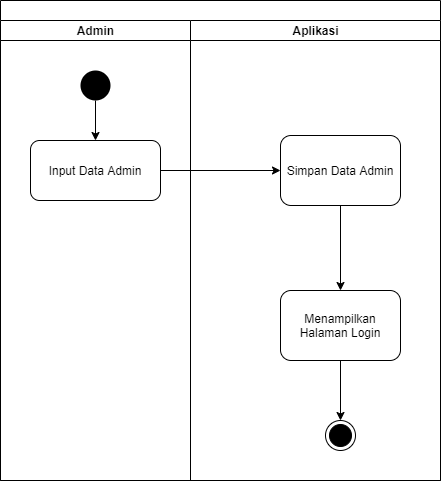
Berikut adalah *activity diagram* melakukan *login* admin dalam bentuk gambar.



Keterangan :

1. Admin menginput *username* dan *password*.
2. Aplikasi melakukan validasi *username* dan *password*.
3. Jika valid akan menampilkan halaman *dashboard*.
4. Jika tidak valid menampilkan pesan *username* dan *password* tidak valid.
5. *Activity diagram* Melakukan Registrasi Admin

Berikut ini merupakan *Activity diagram* Registrasi Admin yang menggambarkan tentang berbagai aktifitas yang terjadi pada suatu sistem. Dimana pada diagram ini, terdiri dari komponen-komponen sebagai berikut.

1. Aktor : Admin dan Aplikasi
2. *Satu initial node*
3. *Activity* : Terdapat tiga *activity* yang dapat dipaparkan sebagai berikut.
4. Input data admin
5. Simpan data admin
6. Menampilkan halaman *login*
7. Satu *Activity* *Final Node*

Keterangan :

1. Admin menginput data admin.
2. Aplikasi menyimpan data admin.
3. Aplikasi menampilkan halaman *login*.
4. *Activity diagram* Melakukan Pengolahan Data

Berikut ini merupakan *Activity diagram* Pengolahan Data yang menggambarkan tentang berbagai aktifitas yang terjadi pada suatu sistem. Dimana pada diagram ini, terdiri dari komponen-komponen sebagai berikut.

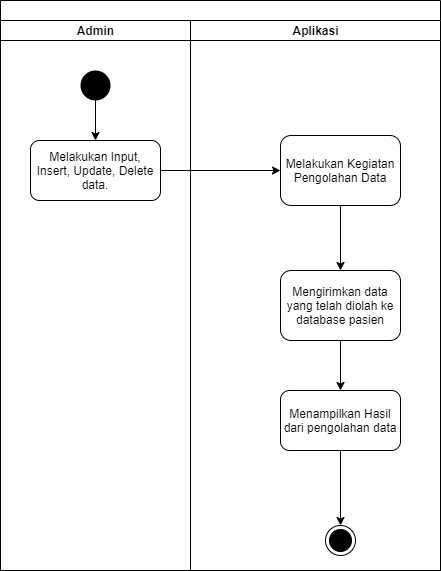
1. Aktor : Admin dan Aplikasi
2. Satu *initial node*
3. *Activity* : Terdapat empat *activity* yang dapat dipaparkan sebagai berikut.
4. Melakukan *insert*, *update*, atau *delete* data
5. Melakukan proses olah data yang dilakukan
6. Mengirim data yang diolah ke *database* pegawai
7. Menampilkan hasil pengolahan data
8. Satu *Activity* *Final Node*

Berikut adalah *activity diagram* melakukan pengolahan data dalam bentuk gambar.

Keterangan :

1. Admin melakukan olah data seperti *insert*, *update*, dan *delete* data.
2. Aplikasi melakukan proses olah data yang dilakukan.
3. Aplikasi mengirim data yang diolah ke *database* pegawai.

Aplikasi menampilkan data hasil create/*update*/*delete*.



1. *Activity diagram* Melakukan Prediksi Kanker Payudara

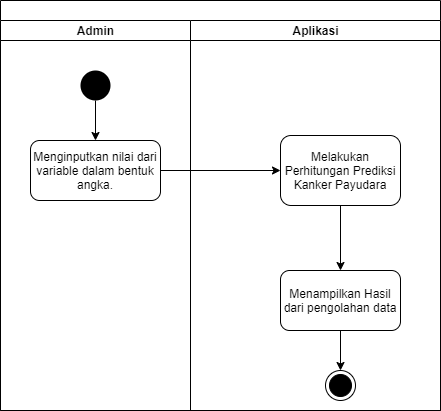
Berikut ini merupakan *Activity diagram* Melakukan Prediksi Kanker yang menggambarkan tentang berbagai aktifitas yang terjadi pada suatu sistem. Dimana pada diagram ini, terdiri dari komponen-komponen sebagai berikut.

1. Aktor : Admin dan Aplikasi
2. Satu *initial node*
3. *Activity* : Terdapat tiga *activity* yang dapat dipaparkan sebagai berikut.

* Menginput variabel independen dalam bentuk angka
* Melakukan perhitungan prediksi kanker
* Menampilkan hasil prediksi kanker

1. Satu *Activity* *Final Node*

Berikut adalah *activity diagram* melakukan prediksi kanket payudara pada pasien dalam bentuk gambar.



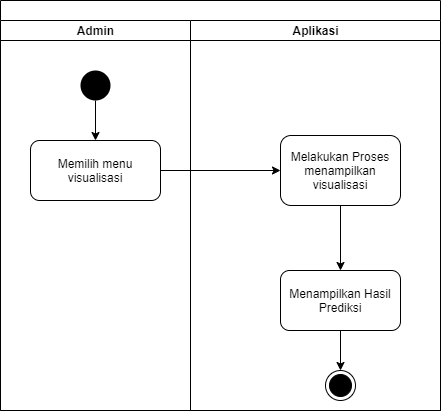
Keterangan :

1. Admin menginputkan nilai dari variabel dalam bentuk angka pada *form* prediksi.
2. Aplikasi melakukan proses perhitungan prediksi kanker.
3. Aplikasi menampilkan hasil prediksi.
4. *Activity diagram* Melakukan Visualisasi

Berikut ini merupakan *Activity diagram* Visualisasi Data yang menggambarkan tentang berbagai aktifitas yang terjadi pada suatu sistem. Dimana pada diagram ini, terdiri dari komponen-komponen sebagai berikut.

1. Aktor : Admin dan Aplikasi
2. Satu *initial node*
3. *Activity* : Terdapat tiga *activity* yang dapat dipaparkan sebagai berikut.
4. Memilih menu visualisasi
5. Melakukan proses menampilkan visualisasi
6. Menampilkan halaman visualisasi
7. Satu *Activity* *Final Node*

Berikut adalah *activity diagram* melakukan pengolahan data dalam bentuk gambar



Keterangan :

1. Admin memilih menu visualisasi.
2. Aplikasi melakukan proses menampilkan visualisasi.
3. Aplikasi menampilkan halamann visualisasi.
4. *Activity diagram* *Logout* Admin

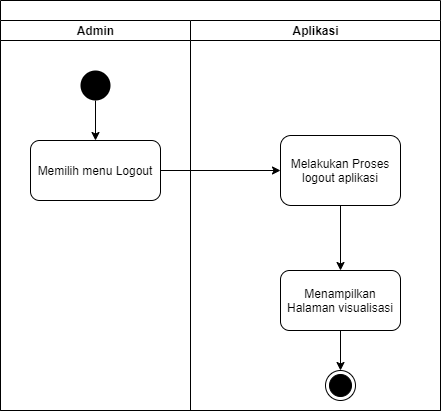
Berikut ini merupakan *Activity diagram* *Logout* Admin yang menggambarkan tentang berbagai aktifitas yang terjadi pada suatu sistem. Dimana pada diagram ini, terdiri dari komponen-komponen sebagai berikut.

1. Aktor : Admin dan Aplikasi
2. Satu *initial node*
3. *Activity* : Terdapat tiga *activity* yang dapat dipaparkan sebagai berikut.
4. Memilih menu *Logout*
5. Melakukan proses *Logout* aplikasi
6. Menampilkan halaman awal aplikasi
7. Satu *Activity* *Final Node*

Keterangan :

* + - 1. Admin memilih menu *Logout*.
      2. Aplikasi melakukan proses *Logout* aplikasi.
      3. Aplikasi menampilkan halaman awal aplikasi.

Berikut adalah *activity diagram* melakukan pengolahan data dalam bentuk gambar.



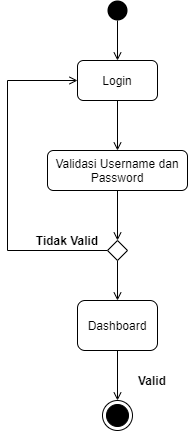
* + 1. **Statechart Diagram**

Diagram statechart dapat digunakan untuk menggambarkan bagaimana suatu objek dalam sistem dapat berpindah dari satu keadaan ke keadaan lainnya sebagai akibat dari stimulus yang diterimanya. Biasanya, diagram statechart dapat digunakan untuk menggambarkan perilaku dari suatu kelas tertentu (satu kelas dapat memiliki lebih dari satu diagram statechart)." Jika ada sesuatu yang tidak jelas, silakan tanyakan kembali. Saya akan senang untuk membantu Anda.

1. *Statechart Diagram* *Login* Admin

Berikut ini merupakan *Activity diagram* *Login* Admin yang menggambarkan transisi dan perubahan keadaan dari suatu objek pada sistem.

Berikut adalah *Statechart Diagram* melakukan *login* admin dalam bentuk gambar.

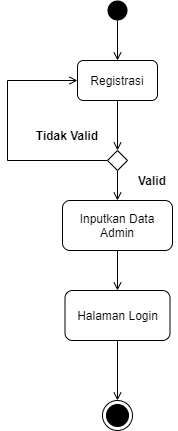
****

Keterangan :

1. Admin melakukan *login*.
2. Aplikasi melakukan validasi *username* dan *password*.
3. Jika valid akan menampilkan halaman *dashboard*.
4. Jika tidak valid akan dikembalikan ke halaman *login*.
5. *Statechart Diagram* Registrasi Admin

Berikut ini merupakan *Activity diagram* Registrasi Admin yang menggambarkan transisi dan perubahan keadaan dari suatu objek pada sistem.

Berikut adalah *Statechart Diagram* melakukan registrasi admin dalam bentuk gambar.

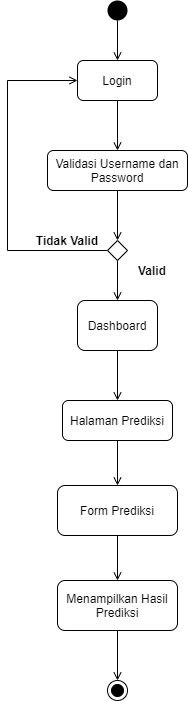
****

Keterangan :

1. Admin menginput data admin.
2. Jika valid maka data admin akan terinput ke *database*.
3. Jika tidak valid maka akan dikembalikan ke halaman registrasi admin.
4. Aplikasi menampilkan halaman *login*.
5. *Statechart Diagram* Melakukan Prediksi

Berikut ini merupakan *Activity diagram* Prediksi Gaji yang menggambarkan transisi dan perubahan keadaan dari suatu objek pada sistem.

Berikut adalah *Statechart Diagram* melakukan prediksi gaji dalam bentuk gambar.

****

Keterangan :

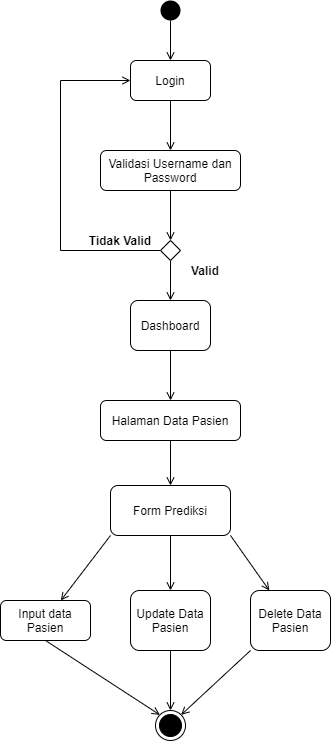
1. Admin melakukan *login*.
2. Aplikasi melakukan validasi *username* dan *password*.
3. Jika valid akan menampilkan halaman *dashboard*.
4. Jika tidak valid akan dikembalikan ke halaman *login*.
5. Aplikasi menampilkan *form* prediksi pada halaman prediksi
6. Aplikasi menampilkan hasil prediksi kanker payudara.
7. *Statechart Diagram* Pengolahan Data

Berikut ini merupakan *Activity diagram* Pengolahan Data Pasien yang menggambarkan transisi dan perubahan keadaan dari suatu objek pada sistem.

Berikut adalah *Statechart Diagram* melakukan pengolahan data dalam bentuk gambar.

Keterangan :

1. Admin melakukan *login*.
2. Aplikasi melakukan validasi *username* dan *password*.
3. Jika valid akan menampilkan halaman *dashboard*.
4. Jika tidak valid akan dikembalikan ke halaman *login*.
5. Aplikasi akan menampilkan halaman data Pasien
6. Pada halaman data karyawan, admin dapat memilih pengolahan antara input data karyawan, *update* data karyawan, ataupun *delete* data Pasien



Keterangan :

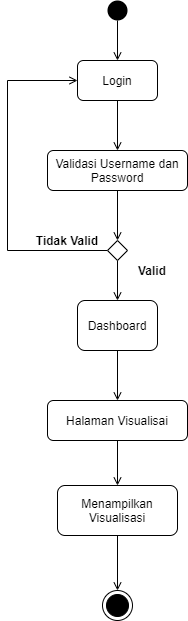
1. Admin melakukan *login*.
2. Aplikasi melakukan validasi *username* dan *password*.
3. Jika valid akan menampilkan halaman *dashboard*.
4. Jika tidak valid akan dikembalikan ke halaman *login*.
5. Aplikasi akan menampilkan halaman data pasien.
6. Pada halaman data pasien, admin dapat memilih pengolahan antara input data karyawan, *update* data karyawan, ataupun *delete* data karyawan
7. *Statechart Diagram* Visualisasi

Berikut ini merupakan *Activity diagram* Visualisasi Data yang menggambarkan transisi dan perubahan keadaan dari suatu objek pada sistem.

Keterangan :

1. Admin melakukan *login*.
2. Aplikasi melakukan validasi *username* dan *password*.
3. Jika valid akan menampilkan halaman *dashboard*.
4. Jika tidak valid akan dikembalikan ke halaman *login*.
5. Aplikasi menampilkan halaman visualisasi.
6. Visualisasi data ditampilkan oleh aplikasi

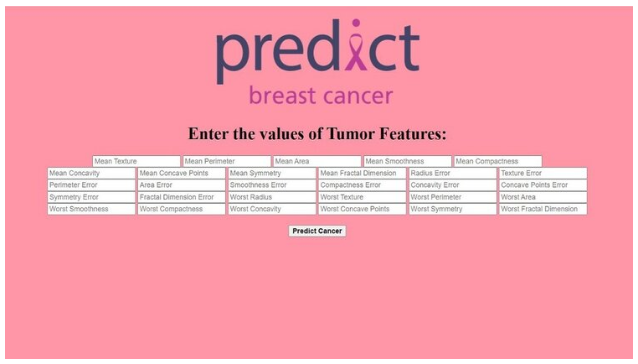
Berikut adalah *Statechart Diagram* melakukan visualisasi data dalam bentuk gambar.

****

* + 1. **Component Diagram**

Diagram komponen dapat menggambarkan struktur dan hubungan antara komponen perangkat lunak, termasuk dependensi di antara mereka. Komponen perangkat lunak atau yang biasa disebut dengan perangkat lunak adalah modul yang berisi kode, baik berisi kode sumber maupun kode biner, baik itu library maupun executable, baik yang muncul pada saat compile time, link time, maupun runtime. Pada umumnya, komponen dapat terdiri dari beberapa kelas dan atau paket, tapi juga dapat dibentuk dari komponen-komponen yang lebih kecil.

<https://www.jietdat.ac.in/acclaimed-projects/predictionofbreastcancer>



Kanker Payudara adalah penyakit paling utama yang mempengaruhi 2,1 juta wanita setiap tahun yang menyebabkan jumlah kematian terbesar di kalangan wanita. Perawatan dini tidak hanya membantu menyembuhkan kanker tetapi juga membantu dalam pencegahan kekambuhannya. Dan karenanya sistem ini terutama berfokus pada prediksi kanker payudara di mana ia menggunakan algoritma pembelajaran mesin yang berbeda untuk membuat model seperti pohon keputusan, regresi logistik, dan hutan acak yang diterapkan pada data pra-proses yang mencurigai akurasi prediksi yang lebih besar. Di antara semua model, Klasifikasi Hutan Acak menghasilkan akurasi terbaik. Teknik-teknik ini dikodekan dalam Python dan menggunakan perpustakaan NumPy, Pandas, Seaborn.

# DAFTAR PUSTAKA

Blogs.reed.edu. 2017. Text analysis using Voyant Tools. [online] Available at: <https://blogs.reed.edu/ed- tech/2017/03/text-analysis-using-voyant-tools/> [Accessed 11 September 2022].

Calado, F., 2018. Using Voyant-Tools to Formulate Research Questions for Textual Data - GC Digital Fellows. [online] GC Digital Fellows. Available at:

<https://digitalfellows.commons.gc.cuny.edu/2018/11/01

/using-voyant-tools-to-formulate-research-questions-for- textual- data/#:~:text=One%20of%20the%20benefits%20of,text

%20is%20displayed%20for%20reading.> [Accessed 11

August 2022].

En.wikipedia.org. 2022. Voyant Tools - Wikipedia. [online] Available at:

<https://en.wikipedia.org/wiki/Voyant\_Tools#:~:text=Ran ge%20of%20Uses-

,History,Lisa%20Goddard%2C%20and%20Mark%20Turc ato.> [Accessed 10 August 2022].

GitHub. 2016. *GitHub - voyanttools/Voyant*. [online] Available at:

<https://github.com/voyanttools/Voyant> [Accessed 10

August 2022].

-oo00oo-

DIHALAMAN INI, SILAHKAN ANDA ISI DENGAN PROFIL PENULIS, BAIK ITU PEMBIMBING INTERNSHP 1 DAN MAHASISWA

BUAT COVER BELAKANG BUKU

YANG BERISI :

SINOPSIS DARI BUKU YANG ANDA TULIS, SINOPSIS DAPAT DIAMBIL DARI ABSTRAK PADA JURNAL YANG ANDA BUAT

CATATAN :

DESAIN HARUS ORIGINAL