

# **MEMBANGUN APLIKASI PREDIKSI JUMLAH KEBUTUHAN PERUSAHAAN**



---

# MEMBANGUN APLIKASI PREDIKSI JUMLAH KEBUTUHAN PERUSAHAAN

---

**Roni Habibi, S.Kom., M.T.**  
**Alwan Suryansah**  
Informatics Research Center



**Kreatif Industri Nusantara**

***Penulis:***

Mohamad Nurkamal Fauzan, S.T., M.T.  
Lalita Chandiany Adiputri

ISBN : xxx-xxx-xxxx-x-2

***Editor:***

M. Yusril Helmi Setyawan

***Penyunting:***

Syafrial Fachrie Pane  
Khaera Tunnisa  
Diana Asri Wijayanti

***Desain sampul dan Tata letak:***

Deza Martha Akbar

***Penerbit:***

Kreatif Industri Nusantara

***Redaksi:***

Jl. Ligar Nyawang No. 2  
Bandung 40191  
Tel. 022 2045-8529  
Email : awangga@kreatif.co.id

***Distributor:***

Informatics Research Center  
Jl. Sariasih No. 54  
Bandung 40151  
Email : irc@poltekpos.ac.id

Cetakan Pertama, 2019

Hak cipta dilindungi undang-undang  
Dilarang memperbanyak karya tulis ini dalam bentuk dan dengan cara  
apapun tanpa ijin tertulis dari penerbit

*'Apa yang terjadi jika  
musik tiada, yah seperti  
itulah hidupmu tanpa  
Ilmu dan Ahklak'  
Alwansah*

# CONTRIBUTORS

---

RONI HABIBI, S.KOM., M.T., ALWAN SURYANSAH,  
Informatics Research Center., Politeknik Pos Indonesia, Bandung, Indonesia



# CONTENTS IN BRIEF

---

<b>1</b>	<b>Pemahaman Aplikasi</b>	<b>1</b>
<b>2</b>	<b>Pengertian Tentang Aplikasi Prediksi</b>	<b>3</b>
<b>3</b>	<b>Perancangan Sistem</b>	<b>63</b>
<b>4</b>	<b>Implementasi</b>	<b>105</b>





# DAFTAR ISI

---

Daftar Gambar	xi
Daftar Tabel	xv
Foreword	xix
Kata Pengantar	xxi
Acknowledgments	xxiii
Acronyms	xxv
Glossary	xxvii
List of Symbols	xxix
Introduction	xxx
<i>Roni Habibi, S.Kom., M.T., Alwan Suryansah.</i>	
<b>1    Pemahaman Aplikasi</b>	<b>1</b>
<b>2    Pengertian Tentang Aplikasi Prediksi</b>	<b>3</b>
2.1    Apa itu Prediksi ?	3
2.1.1    Metode Penggunaan Prediksi	6
	<b>ix</b>

2.2	Penggunaan Bahasa Pemrograman	18
2.2.1	Bahasa Pemograman YII2	18
2.2.2	Sejarah Bahasa YII	21
2.2.3	Fitur - Fitur Yii Framework	22
2.2.4	Kelebihan dan Kekurangan YII Framework	23
2.2.5	Install YII	24
2.3	Penggunaan Database	27
2.3.1	Pengertian Database	27
2.3.2	Pembagian Database	27
2.3.3	Contoh Beberapa Basis Data	28
2.4	Penggunaan XAMPP	40
2.4.1	Pengertian XAMPP	40
2.4.2	Fungsi XAMPP	41
2.4.3	Komponen Penting di Dalam XAMPP	41
2.4.4	Cara MengInstall XAMPP	42
2.5	Penggunaan Aplikasi <i>Text Editor</i>	46
2.5.1	Apa Itu Text Editor?	47
2.5.2	Macam-macam Text Editor	47
2.5.3	Install Submlie Text 3	56

### **3      Perancangan Sistem      63**

3.1	Analisis	63
3.1.1	Pengertian Analisi	63
3.1.2	Analisis Proses Bisnis yang akan dibangun	64
3.1.3	Kebutuhan Aplikasi	65
3.2	Design Aplikasi	65
3.2.1	<i>Use Case diagram</i>	66
3.2.2	Definisi <i>Use Case</i> dan Aktor	68
3.2.3	Skenario <i>Use Case</i>	69
3.2.4	<i>Class diagram</i>	74
3.2.5	<i>Sequence Diagram</i>	76
3.2.6	<i>Activity Diagram</i>	82
3.2.7	Perancangan Antara Muka	91

### **4      Implementasi      105**

4.1	Pembuatan Data Base	105
4.1.1	Perancangan Database	105

# DAFTAR GAMBAR

---

2.1	Proses KDD	7
2.2	Rumus Metode Moving Average	9
2.3	FlowChart Moving Average	10
2.4	contoh Perhitungan <i>bayes</i>	15
2.5	Rumus <i>bayes</i>	15
2.6	Rumus <i>bayes2</i>	16
2.7	Rumus <i>bayes3</i>	16
2.8	Rumus <i>bayes4</i>	16
2.9	Rumus <i>bayes5</i>	16
2.10	<i>Densitas Gauss</i>	17
2.11	<i>Mean</i>	17
2.12	<i>Deviasi Standar</i>	17
2.13	<i>Laplace Correction</i>	18

2.14	MVC	19
2.15	Langkah pertama Install Yii	25
2.16	Langkah Terakhir Install Yii	26
2.17	halaman Yii	26
2.18	Basis Data MySql	29
2.19	Basis Data Oracle	31
2.20	Basis Data Microsoft SQL Server	34
2.21	Basis Data MariaDB	38
2.22	Halaman Download Web Aplikasi Xampp	42
2.23	langkah pertama install xampp	43
2.24	langkah install xampp	44
2.25	menentukan lokasi file disimpan	45
2.26	proses install XAMPP	46
2.27	Logo Sublime	47
2.28	logo Visual Code Studio	49
2.29	Extensions	50
2.30	logo Notepad++	50
2.31	logo Kodomo Edit	52
2.32	Logo Atom	55
2.33	Langkah pertama	57
2.34	Langkah Kedua	57
2.35	Langkah Ketiga	58
2.36	Langkah Keempat	59
2.37	Langkah Kelima	60
2.38	Halaman Sublime Text	61
3.1	FlowChart Analisi yang akan DiBangun	64
3.2	Perangkat Keras	65
3.3	Perangkat Lunak	65

3.4	Usecase	67
3.5	Aktor	67
3.6	Komponen Usecase	68
3.7	Definisi Aktor	68
3.8	Definisi <i>Use Case</i>	69
3.9	Skenario <i>Use Case</i> login	70
3.10	Skenario <i>Use Case</i> Melakukan input Prediksi	71
3.11	Skenario <i>Use Case</i> Cetak laporan hasil Prediksi	71
3.12	Skenario <i>Use Case</i> History Prediksi	72
3.13	Skenario <i>Use Case</i> Kelola Data	73
3.14	Skenario <i>Use Case</i> Logout.	74
3.15	Relasi <i>Class diagram</i>	75
3.16	Macam-Macam Multiplicity	76
3.17	<i>Class diagram</i> Aplikasi Prediksi	76
3.18	<i>Sequence Diagram</i> Login	78
3.19	<i>Sequence Diagram</i> Melakukan Input Prediksi	79
3.20	<i>Sequence Diagram</i> Cetak laporan hasil Prediksi	80
3.21	<i>Sequence Diagram</i> History Prediksi	81
3.22	<i>Sequence Diagram</i> Kelola Data	82
3.23	<i>Start Point/Initial State</i>	83
3.24	<i>Activitiy</i>	84
3.25	<i>Action Flow</i>	84
3.26	<i>Decision</i>	84
3.27	<i>Synchornization</i>	85
3.28	<i>Merge Event</i>	86
3.29	<i>Swimlanes</i>	86
3.30	<i>Final State/End Point</i>	87
3.31	<i>Activity diagram</i> login	87

3.32	<i>Activity diagram</i> Melakukan Input Prediksi	88
3.33	<i>Activity diagram</i> Cetak laporan hasil Prediksi :	89
3.34	<i>Activity diagram History</i> Prediksi	90
3.35	<i>Activity diagram</i> Kelola data	91
3.36	Empat Tahap Proses Perancangan Antarmuka	94
3.37	<i>User Support</i>	97
3.38	Perancangan Antarmuka Halaman Login	98
3.39	Perancangan Antarmuka Halaman Utama	99
3.40	Perancangan Antarmuka Halaman Prediksi	100
3.41	Perancangan Antarmuka Halaman Cetak Hasil Laporan Prediksi	101
3.42	Perancangan Antarmuka Halaman Kelola Data	102
3.43	Perancangan Antarmuka Halaman History	103

# DAFTAR TABEL

---





# Listings

---

Language=PHP

12



# FOREWORD

---

Sepatah kata dari Kaprodi, Kabag Kemahasiswaan dan Mahasiswa



# KATA PENGANTAR

---

Alhamdulillah segala puji dan syukur kehadiran Allah SWT karena dengan rahmat dan karunia-Nya penulis dapat membuat buku sampai saat ini, Dan shalawat serta salam semoga tercurah kepada junjungan Nabi Akhir Zaman, Muhammad SAW.

Buku ini diciptakan untuk menjadi bahan rujuk bagi para pembaca untuk membuat aplikasi prediksi, dan semoga dapat bermanfaat bagi para pembacanya.

Assalamu'alaikum Warahmatullahi Wabarakatuh

RONI HABIBI , ALWAN SURYANSAH

*Bandung, Jawa Barat*

*Januari, 2020*



# ACKNOWLEDGMENTS

---

Terima kasih atas semua masukan dari para mahasiswa agar bisa membuat buku ini lebih baik dan lebih mudah dimengerti.

Terima kasih ini juga ditujukan khusus untuk team IRC yang telah fokus untuk belajar dan memahami bagaimana buku ini mendampingi proses Intership.

R. M. A.





# ACRONYMS

---

ACGIH	American Conference of Governmental Industrial Hygienists
AEC	Atomic Energy Commission
OSHA	Occupational Health and Safety Commission
SAMA	Scientific Apparatus Makers Association



# GLOSSARY

---

- forecasting* adalah suatu proses memperkirakan secara sistematis tentang sesuatu yang paling mungkin terjadi di masa depan berdasarkan informasi masa lalu dan sekarang yang dimiliki, agar kesalahannya (selisih antara sesuatu yang terjadi dengan hasil perkiraan) dapat diperkecil.
- Moving Average* adalah indikator teknikal yang memperhalus pergerakan dengan menyaring fluktuasi yang bersifat acak. Sebagai indikator, *Moving Average* bersifat *trend-following* (mengikuti tren) dan *lagging* (tertunda) karena dilihat berdasarkan yang telah terjadi. Banyak *trader forex* menggunakan *Moving Average* sebagai alat bantu analisa teknikal karena termasuk indikator paling mudah dipakai dan sederhana.



# SYMBOLS

---

$A$  Amplitude

$\&$  Propositional logic symbol

$a$  Filter Coefficient

$\mathcal{B}$  Number of Beats



# INTRODUCTION

---

RONI HABIBI, S.KOM., M.T., ALWAN SURYANSAH.

Informatics Research Center  
Bandung, Jawa Barat, Indonesia

Indonesia saat ini telah memasuki era industri 4.0 atau revolusi 4.0 dimana otomatisasi sistem produksi dengan memanfaatkan teknologi dan big data. sehingga harus ada pengolahan data yang dapat menunjang keberhasilan perusahaan di era 4.0 , dengan melakukan Prediksi untuk jangka pendek, menengah, maupun jangka panjang sangat berpengaruh dalam pengambilan keputusan kedepannya , dengan begitu di dalam buku ini menjelaskan prediksi pada penggunaan kebutuhan .





# BAB 1

---

## PEMAHAMAN APLIKASI

---



## BAB 2

---

# PENGERTIAN TENTANG APLIKASI PREDIKSI

---

### 2.1 Apa itu Prediksi ?

Prediksi adalah suatu proses memperkirakan secara sistematis tentang sesuatu yang paling mungkin terjadi di masa depan berdasarkan informasi masa lalu dan sekarang yang dimiliki, agar kesalahannya (selisih antara sesuatu yang terjadi dengan hasil perkiraan) dapat diperkecil. Prediksi tidak harus memberikan jawaban secara pasti kejadian yang akan terjadi, melainkan berusaha untuk mencari jawaban sedekat mungkin yang akan terjadi (Herdianto, 2013 : 8) .

Pengertian Prediksi sama dengan ramalan atau perkiraan. Menurut kamus besar bahasa Indonesia, prediksi adalah hasil dari kegiatan memprediksi atau meramal atau memperkirakan nilai pada masa yang akan datang dengan menggunakan data masa lalu. Prediksi menunjukkan apa yang akan terjadi pada suatu keadaan tertentu dan merupakan input bagi proses perencanaan dan pengambilan keputusan.

Prediksi bisa berdasarkan metode ilmiah ataupun subjektif belaka. Ambil contoh, prediksi cuaca selalu berdasarkan data dan informasi terbaru yang didasarkan pengamatan termasuk oleh satelit. Begitupun prediksi gempa, gunung meletus ataupun

bencana secara umum. Namun, prediksi seperti pertandingan sepakbola, olahraga, dll umumnya berdasarkan pandangan subjektif dengan sudut pandang sendiri yang memprediksinya.

Permulaan awal, walaupun pengkajian yang mendalam mengenai alternatif masa depan adalah suatu disiplin baru, barangkali orang telah menaruh perhatian besar tentang apa yang akan terjadi kemudian semenjak manusia mulai mengetahui sesuatu. Populasi tukang ramal dan tukang nujum pada zaman kuno dan abad pertengahan merupakan satu manifestasi dari keinginan tahu orang tentang masa depannya. Perhatian tentang masa depan ini berlangsung terus bahkan berkembang menjadi kolom astrologi yang disindikatkan pada tahun 1973.

Secara Eksplisit, pembahasan mengenai teori peramalan kebijakan sangatlah sedikit. Namun, secara implisit, peramalan kebijakan terkait menjadi satu dengan proses analisa kebijakan. Karena didalam menganalisa kebijakan, untuk menformulasikan sebuah rekomendasi kebijakan baru, maka diperlukan adanya peramalan-peramalan atau prediksi mengenai kebijakan yang akan diberlakukan dimasa yang akan datang. Namun, satu dari sekian banyak prosedur yang ditawarkan oleh para pakar Dunn, masih memberikan pembahasan tersendiri mengenai peramalan kebijakan. Menurut Dunn, Peramalan Kebijakan ( policy forecasting) merupakan suatu prosedur untuk membuat informasi factual tentang situasi social masa depan atas dasar informasi yang telah ada tentang masalah kebijakan.

Peramalan (forecasting) adalah suatu prosedur untuk membuat informasi factual tentang situasi sosial masa depan atas dasar informasi yang telah ada tentang masalah kebijakan. Ramalan mempunyai tiga bentuk utama: proyeksi, prediksi, dan perkiraan.

1. Suatu proyeksi adalah ramalan yang didasarkan pada ekstrapolasi atas kecenderungan masa lalu maupun masa kini ke masa depan. Proyeksi membuat pertanyaan yang tegas berdasarkan argument yang diperoleh dari metode tertentu dan kasus yang paralel.
2. Sebuah prediksi adalah ramalan yang didasarkan pada asumsi teoritik yang tegas. Asumsi ini dapat berbentuk hukum teoretis (misalnya hukum berkurangnya nilai uang), proposisi teoritis (misalnya proposisi bahwa pecahnya masyarakat sipil diakibatkan oleh kesenjangan antara harapan dan kemampuan), atau analogi (misalnya analogi antara pertumbuhan organisasi pemerintah dengan pertumbuhan organisme biologis).
3. Suatu perkiraan (conjecture) adalah ramalan yang didasarkan pada penilaian yang informative atau penilaian pakar tentang situasi masyarakat masa depan.

Tujuan dari pada diadakannya peramalan kebijakan adalah untuk memperoleh informasi mengenai perubahan dimasa yang akan datang yang akan mempengaruhi terhadap implementasi kebijakan serta konsekuensinya. Oleh karenanya, sebelum

rekomendasi diformulasikan perlu adanya peramalan kebijakan sehingga akan diperoleh hasil rekomendasi yang benar-benar akurat untuk diberlakukan pada masa yang akan. Didalam memprediksi kebutuhan yang akan datang dengan berpijak pada masa lalu, dibutuhkan seseorang yang memiliki daya sensitifitas tinggi dan mampu membaca kemungkinan-kemungkinan dimasa yang akan datang. Permalan kebijakan juga diperlukan untuk mengontrol, dalam artian, berusaha merencanakan dan menetapkan kebijakan sehingga dapat memberikan alternatif-alternatif tindakan yang terbaik yang dapat dipilih diantara berbagai kemungkinan yang ditawarkan oleh masa depan. Masa depan juga terkadang banyak dipengaruhi oleh masa lalu. Dengan mengacu pada masa depan analisis kebijakan harus mampu menaksir nilai apa yang bisa atau harus membimbing tindakan di masa depan.

Peramalan yang baik mempunyai beberapa kriteria yang penting antara lain sebagai berikut :

1. **Akurasi.** Akurasi dari suatu hasil peramalan diukur dengan kebiasaan dan konsistensi peramalan tersebut. Hasil peramalan dikatakan bias bila peramalan tersebut terlalu tinggi atau terlalu rendah dibandingkan dengan kenyataan yang sebenarnya terjadi. Hasil peramalan dikatakan konsisten bila besarnya kesalahan peramalan relatif kecil. Peramalan yang terlalu rendah, akan mengakibatkan kekurangan persediaan, sehingga permintaan konsumen tidak dapat dipenuhi segera, akibatnya adalah perusahaan dimungkinkan kehilangan pelanggan dan kehilangan keuntungan penjualan. Peramalan yang terlalu tinggi akan mengakibatkan terjadinya penumpukan persediaan, sehingga banyak modal yang terserap sia-sia. Keakuratan dari hasil peramalan ini berperan penting dalam menyeimbangkan persediaan yang ideal, yaitu meminimasi penumpukan persediaan dan memaksimalkan tingkat pelayanan.
2. **Biaya.** Biaya yang diperlukan dalam pembuatan suatu peramalan bergantung kepada jumlah item yang diramalkan, lamanya periode peramalan, dan metode peramalan yang dipakai. Ketiga faktor pemicu biaya tersebut akan mempengaruhi berapa banyak data yang dibutuhkan, Bagaimana pengolahan datanya, yaitu secara manual atau komputerisasi, bagaimana penyimpanan datanya, dan siapa tenaga ahli yang diperbantukan. Pemilihan metode peramalan harus disesuaikan dengan dana yang tersedia dan tingkat akurasi yang ingin didapat, misalnya item-item yang penting akan diramalkan dengan metode yang canggih dan mahal, sedangkan item-item yang kurang penting bisa diramalkan dengan metode yang sederhana dan murah. Prinsip ini merupakan adopsi dari Hukum Pareto (Analisis ABC).
3. **Kemudahan.** Penggunaan metode peramalan yang sederhana, mudah dibuat, dan mudah diaplikasikan, akan memberikan keuntungan bagi perusahaan. Adalah percuma memakai metode yang canggih, tetapi tidak dapat diaplikasikan pada sistem perusahaan karena keterbatasan dana, sumberdaya manusia, maupun peralatan teknologi.

Sifat Hasil Peramalan

Dalam membuat peramalan atau menerapkan hasil suatu peramalan, terdapat beberapa hal yang harus dipertimbangkan, yaitu:

1. Peramalan pasti mengandung kesalahan, artinya peramal hanya bisa mengurangi ketidakpastian yang akan terjadi, tetapi tidak dapat menghilangkan ketidakpastian tersebut.
2. Peramalan seharusnya memberikan informasi tentang berapa ukuran kesalahan, artinya karena peramalan pasti mengandung kesalahan, maka adalah penting bagi peramal untuk menginformasikan seberapa besar kesalahan yang mungkin terjadi.
3. Peramalan jangka pendek lebih akurat dibandingkan peramalan jangka panjang. Hal ini disebabkan pada peramalan jangka pendek, sejumlah faktor yang mempengaruhi permintaan relatif masih konstan, sementara semakin panjang periode peramalan, semakin besar pula kemungkinan terjadinya perubahan faktor yang mempengaruhi permintaan.

### 2.1.1 Metode Penggunaan Prediksi

Kemajuan dalam pengumpulan data dan teknologi penyimpanan yang cepat memungkinkan organisasi menghimpun jumlah data yang sangat luas. Alat dan teknik analisis data yang tradisional tidak dapat digunakan untuk mengekstrak informasi dari data yang sangat besar. Untuk itu diperlukan suatu metoda baru yang dapat menjawab kebutuhan tersebut. Data mining merupakan teknologi yang menggabungkan metoda analisis tradisional dengan algoritma yang canggih untuk memproses data dengan volume besar.

Data mining adalah serangkaian proses untuk menggali nilai tambah dari suatu kumpulan data berupa pengetahuan yang selama ini tidak diketahui secara manual. Kata mining berarti usaha untuk mendapatkan sedikit barang berharga dari sejumlah besar material dasar. Data mining merupakan proses pencarian pola dan relasi-relasi yang tersembunyi dalam sejumlah data yang besar dengan tujuan untuk melakukan klasifikasi, estimasi, prediksi, association rule, clustering, deskripsi dan visualisasi.

Secara garis besar data mining dapat dikelompokkan menjadi 2 kategori utama, yaitu. Descriptive mining, yaitu proses untuk menemukan karakteristik penting dari data dalam suatu basis data. Teknik data mining yang termasuk dalam descriptive mining adalah clustering, association, dan sequential mining.

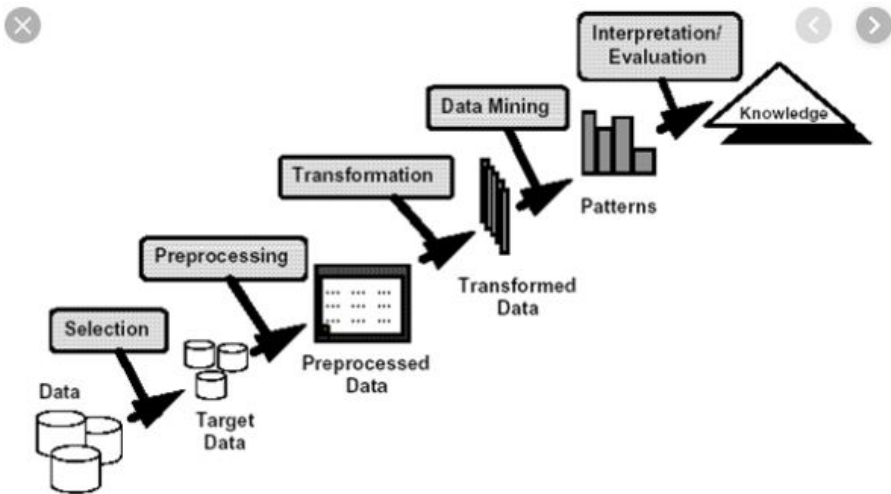
Predictive, yaitu proses untuk menemukan pola dari data dengan menggunakan beberapa variabel lain di masa depan. Salah satu teknik yang terdapat dalam predictive mining adalah klasifikasi. Secara sederhana data mining bisa dikatakan sebagai proses menyaring atau "menambang" pengetahuan dari sejumlah data yang besar. Istilah lain untuk data mining adalah Knowledge Discovery in Database atau KDD. Walaupun sebenarnya data mining sendiri adalah bagian dari tahapan proses dalam KDD

#### *Knowledge Discovery in Database (KDD)*

1. *Definisi*

KDD adalah keseluruhan proses non-trivial agar mencari dan mengidentifikasi pola dalam data, dimana pola yang ditemukan bersifat sah, baru, dapat bermanfaat dan dapat dimengerti.

## 2. Proses KDD



**Gambar 2.1** Proses KDD

Pada proses Knowledge Discovery Database (KDD) terdapat beberapa fase yaitu sebagai berikut:

### (a) *Data Selection*

Pemilihan (seleksi) data dari sekumpulan data operasional perlu dilakukan sebelum tahap penggalian informasi dalam KDD dimulai. Data hasil seleksi yang akan digunakan untuk proses data mining, disimpan dalam suatu berkas, terpisah dari basis data operasional.

### (b) *Data Selection*

Pemilihan (seleksi) data dari sekumpulan data operasional perlu dilakukan sebelum tahap penggalian informasi dalam KDD dimulai. Data hasil seleksi yang akan digunakan untuk proses data mining, disimpan dalam suatu berkas, terpisah dari basis data operasional.

### (c) *Pre-processing/ Cleaning*

Sebelum proses data mining dapat dilaksanakan, perlu dilakukan proses cleaning pada data yang menjadi fokus KDD. Proses cleaning mencakup antara lain membuang duplikasi data, memeriksa data yang inkonsisten, dan memperbaiki kesalahan pada data, seperti kesalahan cetak (tipografi).



Selain itu dilakukan proses enrichment, yaitu proses "memperkaya" data yang sudah ada dengan data atau informasi lain yang relevan dan diperlukan untuk KDD, seperti data atau informasi eksternal.

(d) *Transformation*

Coding adalah proses transformasi pada data yang telah dipilih, sehingga data tersebut sesuai untuk proses Data mining. Proses coding dalam KDD merupakan proses kreatif dan sangat tergantung pada jenis atau pola informasi yang akan dicari dalam basis data, sebagai contoh yaitu jika pada WEKA, mengubah bentuk save as data dari Microsoft Excel Worksheet menjadi CSV (Comma Delimited) sebelum melakukan proses ke data mining.

(e) *Data mining*

Data mining adalah proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik atau metode tertentu. Teknik, metode, atau algoritma dalam data mining sangat bervariasi. Pemilihan metode atau algoritma yang tepat sangat bergantung pada tujuan dan proses KDD secara keseluruhan.

(f) *Interpretation/ Evaluation*

Pola informasi yang dihasilkan dari proses data mining perlu ditampilkan dalam bentuk yang mudah dimengerti oleh pihak yang berkepentingan. Tahap ini merupakan bagian dari proses KDD yang disebut dengan interpretation. Tahap ini mencakup pemeriksaan apakah pola atau informasi yang ditemukan bertentangan dengan fakta atau hipotesa yang ada sebelumnya.

## **Manfaat Data Mining**

Pemanfaatan data mining dapat dilihat dari dua sudut pandang, yaitu sudut pandang komersial dan sudut pandang keilmuan. Dari sudut pandang komersial, pemanfaatan datamining dapat digunakan dalam menangani meledaknya volume data. Bagaimana mana menyimpannya, mengestraknya serta memanfaatkannya. Berbagai teknik komputasi dapat digunakan menghasilkan informasi yang dibutuhkan. Informasi yang dihasilkan menjadi asset untuk meningkatkan daya saing suatu institusi. Data mining tidak hanya digunakan untuk menangani persoalan menumpuknya data/informasi dan bagaimana menggudangkannya tanpa kehilangan informasi yang penting (warehousing). Data mining juga diperlukan untuk menyelesaikan permasalahan atau menjawab kebutuhan bisnis itu sendiri, misalnya:

- (a) Bagaimana mengetahui hilangnya pelanggan karena pesaing
- (b) Bagaimana mengetahui item produk atau konsumen yang memiliki kesamaan karakteristik
- (c) Bagaimana mengidentifikasi produk-produk yang terjual bersamaan dengan produk lain.
- (d) Bagaimana memprediksi tingkat penjualan

- (e) Bagaimana menilai tingkat resiko dalam menentukan jumlah produksi suatu item.
- (f) Bagaimana memprediksi perilaku bisnis di masa yang akan datang Dari sudut pandang keilmuan, data mining dapat digunakan untuk mengcapture, menganalisis serta menyimpan data yang bersifat real time dan sangat besar, misalnya:
  - (a) Remote sensor yang ditempatkan pada suatu satelitP
  - (b) Telescope yang digunakan untuk memindai langit
  - (c) Simulasi saintifik yang membangkitkan data dalam ukuran terabytes

Metode yang di gunakan pada buku ini adalah *Moving Average* yang akan di jelaskan seperti di bawah ini

### 1. *Moving Average*

Rata-rata bergerak (*Moving Average*) adalah suatu metode peramalan yang dilakukan dengan mengambil sekelompok nilai pengamatan, mencari nilai rata-rata tersebut sebagai ramalan untuk periode yang akan datang (Subagyo, 2008). Metode *Moving Average* mempunyai karakteristik khusus yaitu ;

- (a) untuk menentukan ramalan pada periode yang akan datang memerlukan data historis selama jangka waktu tertentu. Misalnya, dengan 3 bulan moving average, maka ramalan bulan ke 5 baru dibuat setelah bulan ke 4 selesai/berakhir. Jika bulan moving averages bulan ke 7 baru bisa dibuat setelah bulan ke 6 berakhir.
- (b) Semakin panjang jangka waktu moving average, efek pelicinan semakin terlihat dalam ramalan atau menghasilkan moving average yang semakin halus.

Persamaan matematis *single moving averages* adalah sebagai berikut:

$$Mt = Ft + 1 \quad (1)$$

$$= \frac{Yt + Yt-1 + Yt-2 + \dots + Yt-n+1}{n} \quad (2)$$

**Gambar 2.2** Rumus Metode Moving Average

Keterangan:

$M_t$  = Moving Average untuk periode  $t$

$F_{t+1}$  = Ramalan Untuk Periode  $t + 1$

$Y_t$  = Nilai Riil periode ke  $t$

$n$  = Jumlah batas dalam moving average

### FlowChar Metode *Moving Average*



**Gambar 2.3** FlowChart Moving Average

## 2. *Time Series* atau Deret Waktu

Analisis *Time Series* merupakan hubungan antara variabel yang dicari (dependent) dengan variabel yang mempengaruhinya (independent variable), yang dikaitkan dengan waktu seperti mingguan, bulan, triwulan, catur wulan, semester atau tahun.

Dalam analisis *time series* yang menjadi variabel yang dicari adalah waktu.

- (a) *Metode Smoting*, merupakan jenis peramalan jangka pendek seperti perencanaan persediaan, perencanaan keuangan. Tujuan penggunaan metode ini adalah untuk mengurangi ketidakteraturan data masa lampau seperti musiman.
- (b) *Metode Box Jenkins*, merupakan deret waktu dengan menggunakan model matematis dan digunakan untuk peramalan jangka pendek.
- (c) *Metode proyeksi trend dengan regresi*, merupakan metode yang digunakan baik untuk jangka pendek maupun jangka panjang. Metode ini merupakan garis trend untuk persamaan matematis.

### 3. **Causal Methods atau sebab akibat**

Merupakan metode peramalan yang didasarkan kepada hubungan antara variabel yang diperkirakan dengan variabel lain yang mempengaruhinya tetapi bukan waktu. Dalam prakteknya jenis metode peramalan ini terdiri dari :

- (a) *Metode regresi* dan kolerasi, merupakan metode yang digunakan baik untuk jangka panjang maupun jangka pendek dan didasarkan kepada persamaan dengan teknik least squares yang dianalisis secara statis.
- (b) *Model Input Output*, merupakan metode yang digunakan untuk peramalan jangka panjang yang biasa digunakan untuk menyusun trend ekonomi jangka panjang.
- (c) *Model ekonometri*, merupakan peramalan yang digunakan untuk jangka panjang dan jangka pendek.

#### **Metode regresi**

Penggunaan metode ini didasarkan kepada variabel yang ada dan yang akan mempengaruhi hasil peramalan.

Hal- hal yang perlu diketahui sebelum melakukan peramalan dengan metode regresi adalah mengetahui terlebih dahulu mengetahui kondisi- kondisi seperti :

- (a) Adanya informasi masa lalu
- (b) Informasi yang ada dapat dibuatkan dalam bentuk data (dikuantifikasikan)
- (c) Diasumsikan bahwa pola data yang ada dari data masa lalu akan berkelanjutan dimasa yang akan datang.

Adapun data- data yang ada dilapangan adalah :

- (a) Musiman (Seasonal)
- (b) Horizontal (Stationary)
- (c) Siklus (Cylikal)
- (d) Trend

Dalam menyusun ramalan pada dasarnya ada 2 macam analisis yang dapat digunakan yaitu :

- (a) Analisi deret waktu(Time series), merupakan analisis antaravariabel yang dicari dengan variabel waktu
- (b) Analisis Cross Section atau sebab akibat (Causal method), merupakan analisis variabel yang dicari dengan variabel bebas atau yang mempengaruhi.

Ada dua pendekatan untuk melakukan peramalan dengan menggunakan analisis deret waktu dengan metode regresi sederhana yaitu :

- (a) Analisis deret waktu untuk regresi sederhana linier
- (b) Analisis deret untuk regresi sederhana yang non linier

Untuk menjelaskan hubungan kedua metode ini kita gunakan notasi matematis seperti :

$$Y = F (x)$$

Dimana :

Y = Dependent variable (variabel yang dicari)

X = Independent variable (variabel yang mempengaruhinya)

### **ANALISIS DENGAN *REGRESI LINIER CROSS SECTION***

Cross section method atau casual method atau sebab akibat merupakan peramalan yang kita lakukan untuk mengukur peramalan dalam suatu periode dengan faktor yang mempengaruhinya bukan waktu.

Penggunaan rumusan yang kita gunakan untuk cross section sama dengan penerapan untuk metode time series, begitu puka dngan hasil pramalannya.

- 1 Jadi penjualan = f (x, x, x, .)
- 2 X = harga, mutu pendapatan, promosi dll
- 3  $Y = a + b x$
- 4
- 5 Dimana x adalah variabel bukan waktu.

## ANALISIS DERET WAKTU DENGAN *REGRESI LINIER*

Ada 2 pendekatan untuk melakukan peramalan dengan menggunakan analisis deret waktu dengan metode regresi sederhana, yaitu :

- (a) Analisis deret waktu untuk regresi sederhana linier
- (b) Analisis deret waktu untuk regresi sederhana yang non linier

Dalam analisis deret waktu yang linier adalah analisis pola hubungan yang dicari dengan satu variabel yang mempengaruhinya : waktu. Sedangkan analisis deret waktu yang non linier, merupakan analisis hubungan antara variabel yang dicari dengan hanya satu (1) yang mempengaruhinya, yaitu variabel waktu.

Untuk menjelaskan hubungan kedua metode ini kita gunakan notasi matematis seperti :

$$Y = F(x)$$

Dimana :

Y = Dependent variable (variabel yang dicari)

X = Independent variable (variabel yang mempengaruhinya)

Notasi regresi sederhana dengan menggunakan regresi linier (garis lurus) dapat digunakan sebagai berikut :

$$Y = a + b X \quad \dots$$

Dimana a dan b adalah merupakan parameter (koefisien regresi) yang harus dicari. Untuk mencari nilai a dapat digunakan dengan menggunakan rumus :

Langkah selanjutnya adalah melakukan pengujian terhasi yang diperoleh dengan :

- (a) uji Test Koefisien Penentu ( $R^2$ ), pengtesan ini untuk mengetahui tepat tidaknya variabel yang mempengaruhi besarnya penjualan yang diramalkan adalah waktu.

Kedua, Test Significance (T.Test) atau F test yaitu pengtesan untuk mengetahui apakah benar persamaan regresi itu adalah linier.

## ANALISA DERET WAKTU DENGAN *REGRESI NON LINIER*

Analisa deret waktu dengan regresi non linier merupakan regresi bukan garis lurus. Notasi regresi sederhana dengan menggunakan regresi linier (garis lurus) dapat digunakan sebagai berikut :

$$Y = a + b x + c x^2$$

Dimana : Y = Dependent variable (variabel yang dicari)

x = Independent variable (variabel yang mempengaruhinya)

a = b = c = parameter koefisien regresi

Formula umum yang digunakan sebagai berikut :

1	y	=	n	a	+	b	x	+	c	x <sup>2</sup>
2	x y	=	a	x	+	b	x <sup>2</sup>	+	c	x <sup>3</sup>
3	x y	=	a	x <sup>2</sup>	+	b	x <sup>3</sup>	+	c	x <sup>4</sup>

#### 4. Algoritma Naive Bayes

Bayesian classification adalah pengklasifikasian statistik yang dapat digunakan untuk memprediksi probabilitas keanggotaan suatu class. Bayesian classification didasarkan pada teorema Bayes yang memiliki kemampuan klasifikasi serupa dengan decision tree dan neural network. Bayesian classification terbukti memiliki akurasi dan kecepatan yang tinggi saat diaplikasikan ke dalam database dengan data yang besar. (Kusrini,2009).

sebuah metoda klasifikasi menggunakan metode probabilitas dan statistik yg dikemukakan oleh ilmuwan Inggris Thomas Bayes. Algoritma Naive Bayes memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dikenal sebagai Teorema Bayes. Ciri utama dr Nave Bayes Classifier ini adalah asumsi yg sangat kuat (naf) akan independensi dari masing-masing kondisi / kejadian.

*Naive Bayes Classifier* bekerja sangat baik dibanding dengan model *classifier* lainnya. Hal ini dibuktikan pada jurnal *Xhemali, Daniela, Chris J. Hinde, and Roger G. Stone. Naivebayes vs. decision trees vs. neural networks in the classification of training web pages.* (2009), mengatakan bahwa *Nave Bayes Classifier* memiliki tingkat akurasi yg lebih baik dibanding model *classifier* lainnya.

Keuntungan peggungan adalah bahwa metoda ini hanya membutuhkan jumlah data pelatihan (*training data*) yang kecil untuk menentukan estimasi parameter yg diperlukan dalam proses pengklasifikasian. Karena yg diasumsikan sebagai variabel independent, maka hanya varians dari suatu variabel dalam sebuah kelas yang dibutuhkan untuk menentukan klasifikasi, bukan keseluruhan dari matriks kovarians.

**Tahapan dari proses algoritma Naive Bayes adalah:**

- (a) Menghitung jumlah kelas / label.
- (b) Menghitung Jumlah Kasus Per Kelas
- (c) Kalikan Semua Variable Kelas
- (d) Bandingkan Hasil Per Kelas

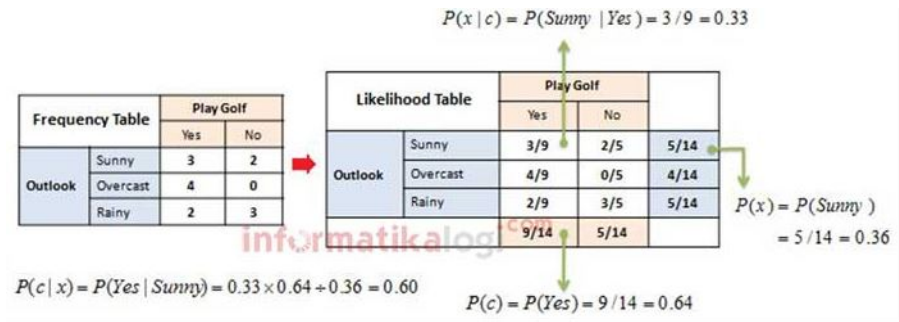
**Kelebihan Kekurangan Naive Bayes**  
**kelebihan**

- (a) Mudah untuk dibuat
- (b) Hasil bagus

**Kekurangan**

- (a) Asumsi independence antar atribut membuat akurasi berkurang (karena biasanya ada keterkaitan)

### Persamaan *Teorema Bayes*



**Gambar 2.4** contoh Perhitungan *bayes*

$$P(C|X) = \frac{P(x|c)P(c)}{P(x)}$$

likelihood points to  $P(x|c)$   
Class Prior Probability points to  $P(c)$   
Posterior Probability points to  $P(C|X)$   
Predictor Prior Probability points to  $P(x)$

**Gambar 2.5** Rumus *bayes*

Keterangan :

$x$  : Data dengan class yang belum diketahui

$c$  : Hipotesis data merupakan suatu class spesifik

$P(c|x)$  : Probabilitas hipotesis berdasar kondisi (posteriori probability)

$P(c)$  : Probabilitas hipotesis (prior probability)

$P(x|c)$  : Probabilitas berdasar kondisi pada hipotesis

$P(x)$  : Probabilitas  $c$

Rumus diatas menjelaskan bahwa peluang masuknya sampel karakteristik tertentu dalam kelas  $C$  (Posterior) adalah peluang munculnya kelas  $C$  (sebelum masuknya sampel tersebut, seringkali disebut prior), dikali dengan peluang kemunculan karakteristik karakteristik sampel pada kelas  $C$  (disebut juga likelihood), dibagi dengan peluang kemunculan karakteristik sampel secara global (disebut juga evidence). Karena itu, rumus diatas dapat pula ditulis sebagai berikut :



$$\text{posterior} = \frac{\text{prior} \times \text{likelihood}}{\text{evidence}}$$

**Gambar 2.6** Rumus bayes2

Nilai Evidence selalu tetap untuk setiap kelas pada satu sampel. Nilai dari posterior tersebut nantinya akan dibandingkan dengan nilai nilai posterior kelas lainnya untuk menentukan ke kelas apa suatu sampel akan diklasifikasikan. Penjabaran lebih lanjut rumus Bayes tersebut dilakukan dengan menjabarkan  $(c \rightarrow x_1, \dots, x_n)$  menggunakan aturan perkalian sebagai berikut :

$$\begin{aligned} P(C|X_1, \dots, X_n) &= P(C)P(X_1, \dots, X_n|C) \\ &= P(C)P(X_1|c)(X_2, \dots, X_n|C, X_1) \\ &= P(C)P(X_1|c)P(X_2|C, X_1)(X_3, \dots, X_n|C, X_1, X_2) \\ &= P(C)P(X_1|c)P(X_2|C, X_1)P(X_3|C, X_1, X_2) \dots P(X_n|C, X_1, X_2, \dots, X_{n-1}) \end{aligned}$$

**Gambar 2.7** Rumus bayes3

Dapat dilihat bahwa hasil penjabaran tersebut menyebabkan semakin banyak dan semakin kompleksnya faktor faktor syarat yang mempengaruhi nilai probabilitas, yang hampir mustahil untuk dianalisa satu persatu. Akibatnya, perhitungan tersebut menjadi sulit untuk dilakukan. Disinilah digunakan asumsi independensi yang sangat tinggi (naif), bahwa masing masing petunjuk saling bebas (independen) satu sama lain. Dengan asumsi tersebut, maka berlaku suatu kesamaan sebagai berikut:

$$P(c|X_1, \dots, X_n) = P(C) \prod_{i=1}^n P(X_i|C)$$

**Gambar 2.8** Rumus bayes4

$$P(c|X) = P(x_1|c)P(x_2|c) \dots P(x_n|c)P(c)$$

**Gambar 2.9** Rumus bayes5

Persamaan diatas merupakan model dari Teorema Naive Bayes yang selanjutnya akan digunakan dalam proses klasifikasi. Untuk klasifikasi dengan data kontinyu digunakan rumus Densitas Gauss :

$$P = (X_i = x_i | Y_i = y_i) = \frac{1}{\sqrt{2\pi\sigma_{ij}^2}} e^{-\frac{(x_i - \mu_{ij})^2}{2\sigma_{ij}^2}}$$

**Gambar 2.10** Densitas Gauss

Keterangan :

P : Peluang

Xi : Atribut ke i

xi : Nilai atribut ke i

Y : Kelas yang dicari

yj : Sub kelas Y yang dicari

u : Mean, menyatakan rata rata dari seluruh atribut

o : Deviasi standar, menyatakan varian dari seluruh atribut

### Mean

$$\mu = \frac{1}{n} \sum_{i=1}^n x_i$$

**Gambar 2.11** Mean

### Deviasi Standar

$$\sigma = \left[ \frac{1}{n-1} \sum_{i=1}^n (x_i - \mu)^2 \right]^{0.5}$$

**Gambar 2.12** Deviasi Standar

### ***Laplace Correction***

*Laplace Correction* (Laplacian Estimator) atau *additive smoothing* adalah suatu cara untuk menangani nilai probabilitas 0 (nol). Dari sekian banyak data di training set, pada setiap perhitungan datanya ditambah 1 (satu) dan tidak akan membuat perbedaan yang berarti pada estimasi probabilitas sehingga bisa menghindari kasus nilai probabilitas 0 (nol).

$$\rho_i = \frac{m_i + 1}{n + k}$$

**Gambar 2.13** *Laplace Correction*

dimana nilai k adalah jumlah kelas atau bin dari atribut  $m_i$ .

## **2.2 Penggunaan Bahasa Pemrograman**

### **2.2.1 Bahasa Pemrograman Yii2**

Yii ialah kerangka kerja PHP berbasis-komponen dengan performansi untuk pengembangan aplikasi atau Web berskala-besar. menyediakan resuabilitas maksimum dalam pemrograman aplikasi atau Web dan bisa mengakselerasi proses pengembangan secara signifikan. Nama Yii (dieja sebagai /i:/) singkatan dari easy, efficient dan extensible (mudah, efisien, dan bisa diperluas) atau singkatan dari "Yes It Is!". Seperti juga kerangka kerja PHP pada umumnya, Yii juga telah mengadopsi konsep MVC - Model, View, Controller dalam struktur pemrogramannya. Untuk menjalankan aplikasi Web bertenaga-Yii, memerlukan server Web yang mendukung PHP 5.1.0 atau lebih tinggi.

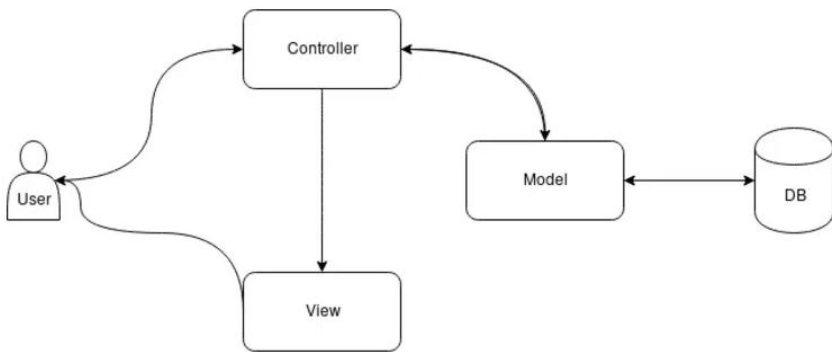
Bagi para pengembang yang ingin menggunakan Yii, mengerti pemrograman terorientasi-obyek (OOP) akan sangat membantu, karena Yii murni kerangka kerja OOP.

Yii adalah kerangka kerja pemrograman umum Web yang bisa dipakai untuk mengembangkan semua jenis aplikasi Web. Oleh karena Yii sangat ringan dan dilengkapi dengan solusi caching yang memuaskan, ia sangat cocok untuk pengembangan aplikasi dengan lalu lintas-tinggi, seperti portal, forum, sistem manajemen konten (CMS), sistem e-commerce, dll.

MVC adalah konsep arsitektur dalam pembangunan aplikasi berbasis web yang membagi aplikasi web menjadi 3 bagian besar. Yang mana setiap bagian memi-

liki tugas-tugas serta tanggung jawab masing-masing. Tiga bagian tersebut adalah: model, view dan controller.

1. Model Bertugas untuk mengatur, menyiapkan, memanipulasi dan mengorganisasikan data (dari database) sesuai dengan instruksi dari controller.
2. View Bertugas untuk menyajikan informasi (yang mudah dimengerti) kepada user sesuai dengan instruksi dari controller.
3. Controller Bertugas untuk mengatur apa yang harus dilakukan model, dan view mana yang harus ditampilkan berdasarkan permintaan dari user. Namun, terkadang permintaan dari user tidak selalu memerlukan aksi dari model. Misalnya seperti menampilkan halaman form untuk registrasi user.



**Gambar 2.14** MVC

#### Framework yang Menggunakan Konsep MVC

Untuk penggunaan framework PHP, ada cukup banyak yang menggunakan konsep MVC seperti :

1. Laravel.
2. Symfony.
3. CakePHP.
4. Zend,
5. Codeigniter (versi 3 kebawah sudah tidak recommended untuk dipelajari).
6. dan lain-lain.

Untuk Penggunaan framework Python di antaranya:

1. Django.

2. Turbogears2.
3. Watson-Framework.
4. dan lain-lain.

Untuk Penggunaan framework Nodejs di antaranya:

1. Express.
2. Adonis.
3. Sails.js
4. Total.js
5. Mean.js
6. Mojito
7. dan lain-lain.

OOP (Object Oriented Programming) iyalah metode pemrograman yang berorientasi kepada objek. Tujuan dari OOP diciptakan adalah untuk mempermudah pengembangan program dengan cara mengikuti model yang telah ada di kehidupan sehari-hari. Jadi setiap bagian dari suatu permasalahan adalah objek, nah objek itu sendiri merupakan gabungan dari beberapa objek yang lebih kecil lagi. Saya ambil contoh Pesawat, Pesawat adalah sebuah objek. Pesawat itu sendiri terbentuk dari beberapa objek yang lebih kecil lagi seperti mesin, roda, baling-baling, kursi, dll. Pesawat sebagai objek yang terbentuk dari objek-objek yang lebih kecil saling berhubungan, berinteraksi, berkomunikasi dan saling mengirim pesan kepada objek-objek yang lainnya. Begitu juga dengan program, sebuah objek yang besar dibentuk dari beberapa objek yang lebih kecil, objek-objek itu saling berkomunikasi, dan saling mengirim pesan kepada objek yang lain.

Konsep OOP terdiri dari Abstrak Class, Encapsulation, Inheritance, Polymorphism :

#### **Abstark Class**

1. Kelas merupakan deskripsi abstrak informasi dan tingkah laku dari sekumpulan data.
2. Kelas dapat diilustrasikan sebagai suatu cetak biru(blueprint) atau prototipe yang digunakan untuk menciptakan objek.
3. Kelas merupakan tipe data bagi objek yang mengenkapsulasi data dan operasi pada data dalam suatu unit tunggal.
4. Kelas mendefinisikan suatu struktur yang terdiri atas data kelas (data field), prosedur atau fungsi (method), dan sifat kelas (property).

## Encapsulation

1. Istilah enkapsulasi sebenarnya adalah kombinasi data dan fungsionalitas dalam sebuah unit tunggal sebagai bentuk untuk menyembunyikan detail informasi.
2. Proses enkapsulasi memudahkan kita untuk menggunakan sebuah objek dari suatu kelas karena kita tidak perlu mengetahui segala hal secara rinci.
3. Enkapsulasi menekankan pada antarmuka suatu kelas, atau dengan kata lain bagaimana menggunakan objek kelas tertentu.
4. Contoh: kelas mobil menyediakan antarmuka fungsi untuk menjalankan mobil tersebut, tanpa kita perlu tahu komposisi bahan bakar, udara dan kalor yang diperlukan untuk proses tersebut.

## Inheritance

1. Kita dapat mendefinisikan suatu kelas baru dengan mewarisi sifat dari kelas lain yang sudah ada.
2. Penurunan sifat ini bisa dilakukan secara bertingkattingkat, sehingga semakin ke bawah kelas tersebut menjadi semakin spesifik.
3. Sub kelas memungkinkan kita untuk melakukan spesifikasi detail dan perilaku khusus dari kelas supernya.
4. Dengan konsep pewarisan, seorang programmer dapat menggunakan kode yang telah ditulisnya pada kelas super berulang kali pada kelas-kelas turunannya tanpa harus menulis ulang semua kodekode itu.

## Polymorphism

1. Polimorfisme merupakan kemampuan objekobjek yang berbeda kelas namun terkait dalam pewarisan untuk merespon secara berbeda terhadap suatu pesan yang sama.
2. Polimorfisme juga dapat dikatakan kemampuan sebuah objek untuk memutuskan method mana yang akan diterapkan padanya, tergantung letak objek tersebut pada jenjang pewarisan.
3. Method overriding.
4. Method name overloading.

### 2.2.2 Sejarah Bahasa Yii

Pengembangan Yii dimulai sejak 1 Januari 2008 oleh Qiang Xue programmer asal Tiongkok. Pada awalnya Yii dikembangkan menggunakan bahasa kerangka kerja PRADO, setelah mengalami beberapa koreksi dan penyempurnaan kurang lebih setahun, pada 3 Desember 2008, Yii 1.0 secara resmi dirilis ke publik. yang pada awalnya bertujuan untuk memperbaiki beberapa kekurangan yang ada pada Framework

PRADO. Pada versi awal framework PADRO berjalan lambat ketika menangani halaman - halaman yang kompleks. Pada waktu itu framework PADRO kurang diminati karena banyak kontrol yang sulit untuk melakukan penyesuaian dengan kebutuhan.

pada suatu ketika Qiang Xue mencoba membangun Yii framework yang lebih stabil, selain itu dia juga mengadopsi teknik-teknik yang dipakai pada Prado untuk mengembangkan Yii. Yii ini terinspirasi oleh PADRO, dan Yii juga terinspirasi oleh framework luar yaitu Ruby On Rails terutama pada mekanisme ORM nya. Disamping itu masih banyak lagi yang menginspirasi Yii framework seperti Symfony dan Joomla. Untuk versi alpha secara resmi Yii dirilis pada tanggal 3 Desember 2008 yaitu Yii 1.0 dan sampai sekarang Yii sudah mencapai versi v1.1.12 yang dirilis pada bulan 19 Agustus 2012.

### 2.2.3 Fitur - Fitur Yii Framework

Dalam menunjang keberhasilan sebuah proyek dengan menggunakan menggunakan bahasa pemrograman YII, dengan begitu tersedianya fitur - fitur YII Framework seperti :

#### 1. MVC (Model View Controller)

Pada framework Yii menggunakan arsitektur programing MVC yaitu menggunakan konsep Model, View, Controller. Hampir seluruh framework PHP menggunakan konsep MVC termasuk CMS (Content management system).

#### 2. ORM(Object Relational Mapping)

Salah satu fitur di Yii framework adalah Object Relational Mapping atau ORM yang mengurus pengelolaan database, jika kita lihat secara spesifik maka Yii sebenarnya menyediakan mekanisme seperti DAO (Data Access Object), Query Builder, Active Record (ORM) dan Database Migration. Dengan ORM pekerjaan developer menjadi terbantu karena tidak lagi menulis query SQL yang rumit, cukup dengan memanggil fungsi-fungsi tertentu di dalam library Yii.

#### 3. Form input dan Validasi

Untuk pembuatan form Yii framework mempunyai fitur form input dan validasi form.

#### 4. Widget

Yii framework mempunyai fitur widget. Widget merupakan konsep Yii yang menyediakan komponen-komponen user interface yang siap pakai, misalnya data grid, autocomplete, tree view dan lain-lain. Sebagian widget-widget ini ada yang terintegrasi dengan library JQuery sehingga memungkinkan untuk mekanisme AJAX.

#### 5. Extension

Yii framework memiliki segudang extensions. Extension merupakan kontribusi dari pengguna-pengguna Yii dengan membuat widget, library ataupun component yang bisa dipakai dan belum ada di Yii secara built-in.

## 6. **Autentikasi dan Autorisasi**

Yii framework mempunyai fitur autentikasi hak akses dan otorisasi logging dan logout.

## 7. **Thema**

Untuk tampilan (view) Yii framework menggunakan konsep skinning kita bisa meng-costumize tampilan komponen user interface pada aplikasi menggunakan CSS. Sedangkan dengan konsep theming kita bisa merubah seluruh tampilan aplikasi.

## 8. **Security**

Untuk fasilitas keamanan Yii mempunyai security yang dapat mencegah serangan-serangan seperti SQL Injection, XSS, CSRF, Cookie Tampering dll.

## 9. **Web service**

Yii framework mempunyai fasilitas Web service sehingga dapat meng-generate spesifikasi WSDL service secara otomatis.

## 10. **Translate**

Yii menyediakan fitur translate (penterjemah) untuk pesan, tanggal dan format waktu sesuai zona waktu dan daerah.

## 11. **Caching**

Yii mempunyai fitur caching sehingga aplikasi yang dibangun menggunakan Yii akan berjalan ringan ketika diakses.

## 12. **Autentikasi error**

Yii menyediakan pesan error dan logging yang bisa memilih untuk menampilkan langsung ataupun disimpan ke suatu file di server.

## 13. **3rd Part (Pihak ketiga)**

Pada Yii framework di desain sedemikian rupa agar kita bisa memasang library external dari pihak ketiga untuk di sematkan dalam aplikasi, kita bisa memasang teknologi bootstrap untuk teknologi tampilan yii.

## 14. **Tutorial**

Yii framework mempunyai dokumentasi yang bisa digunakan untuk mempelajari cara kerja yii sehingga mempermudah membuat aplikasi yang diinginkan.

## 2.2.4 **Kelebihan dan Kekurangan YII Framework**

### **KELEBIHAN YII Framework**

Di awal tahun 2018, YII semakin dipopulerkan oleh beberapa developer. Lalu apa sih kelebihan framework ? Berikut beberapa kelebihannya:

1. **Lebih mudah dan praktis.** Banyak fungsi yang dapat kamu panggil pada framework ini. Sebut saja fungsi GII yang dapat mempermudahmu membuat CRUD dengan sekali jalan.



2. **Kemanan yang terjamin.** Yii Framework telah dibekali dengan berbagai macam kemanan website, sebut saja tools untuk mencegah dan mengatasi XSS attacks, CSRF attacks, serta SQL injections. Ada pula fitur pendukung kemanan lainnya, seperti input validation standard output filtering, dan fitur hash password untuk mengenkripsi dan mendekripsi password saat login.
3. **Caching yang mumpuni.** Kelebihan lainnya dari Yii ialah kemampuan untuk melakukan caching secara optimal. Pada Yii, caching dilakukan untuk membuat website dinamis menjadi seolah website statis, sehingga kecepatan akses pada website akan lebih cepat.
4. **Cocok untuk projek dengan skala menengah keatas dan memiliki keterbatasan waktu pembuatan.**

### **KEKURANGAN Yii Framework**

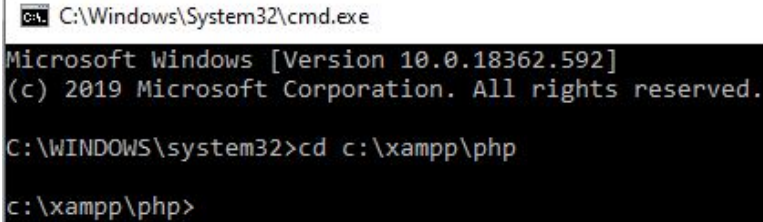
Tiada gading yang tak retak, seperti halnya juga pada jenis Kerangka Kerja PHP ini. Yii Memiliki beberapa kelemahan yang agak mengganggu developer. Apa saja kekurangannya? Simak ulasan berikut:

1. **Developer lebih tidak peduli terhadap keamanan.** Mengapa demikian? Karena kemudahan dalam mengamankan website telah dibebankan sepenuhnya pada sistem Yii. Karenanya, fokus developer di bidang keamanan akan lebih sedikit.
2. **Framework yang terbilang jadul.** Dibuat pada tahun 2006, Yii tergolong telah berusia lebih dari 12 tahun. Seiring perkembangan zaman, serta perkembangan bahasa pemrograman, Yii dikhawatirkan tidak bisa mengikuti perkembangan tersebut.
3. **Bukan merupakan framework yang tercepat.** Jika dibandingkan dengan framework terkenal lainnya, Yii2 (393 rps) duduk diperingkat ke dua dibawah CodeIgniter (565 rps), lalu posisi selanjutnya ada Zend2 (99 rps), Laravel5 (65 rps) serta yang terakhir ada Symfony(6 rps).
4. **Yii bukan framework terpopuler.** Walaupun tak semua yang populer itu mencerminkan banyak kelebihan, namun setidaknya ketika developer mengalami masalah, ada banyak orang yang dapat diajak ikut berdiskusi.

## **2.2.5 Install Yii**

langkah-langkah untuk menginstall Yii2 via composer seperti sebagai berikut:

1. Langkah Pertama buka command prompt
2. setelah itu Asumsi pake xampp di drive c, masuk ke folder yang ada php.exe, Ketik aja



```

C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.18362.592]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>cd c:\xampp\php
c:\xampp\php>

```

**Gambar 2.15** Langkah pertama Install YII

3. lalu ketik seperti di bawah ini

```
1 php.exe -r "readfile('https://getcomposer.org/installer');" | php
   .exe
```

kalo sukses akan muncul!usr/bin/env php All settings correct for using Composer DownloadingComposer successfully installed to: C:.phar Use it: php composer.phar

4. Update composer (cara terbaik untuk mendapatkan versi terbaru dari composer) , ketik aja

```
1 php.exe composer.phar self-update
```

5. Kalo udah.. silahkan buat project baru.. Yii2 dan letakkan di htdocs misal namanya advanced, ketik

```
1 php.exe composer.phar global require "fxp/composer-asset-plugin
   :1.0.0-beta3"
```

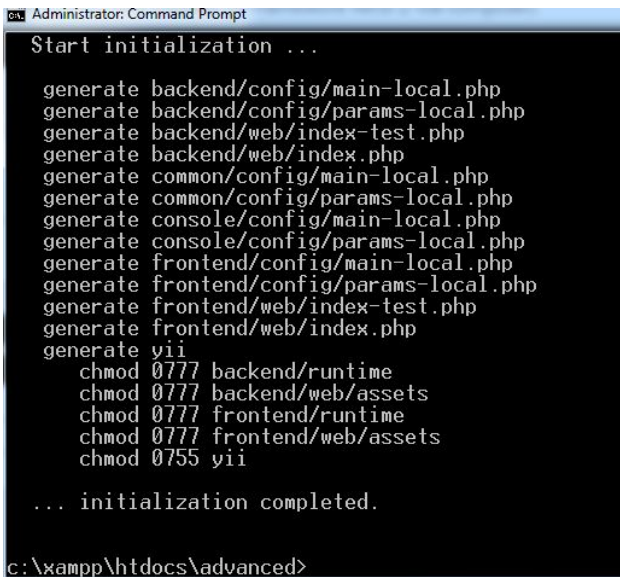
lalu kemudian

```
1 php.exe composer.phar create-project --prefer-dist --stability=
   dev yiisoft/yii2-app-advanced ../htdocs/advanced
```

6. Lalu inisialisasi project, dengan cara ketik

```
1 php.exe ../htdocs/advanced/init
2
```

7. Lalu akan ada pilihan 0 untuk Development or 1 untuk Production, Pilih aja 0 lalu enter, Lalu ada pertanyaan lagi.. ketik yes



```
Administrator: Command Prompt

Start initialization ...

generate backend/config/main-local.php
generate backend/config/params-local.php
generate backend/web/index-test.php
generate backend/web/index.php
generate common/config/main-local.php
generate common/config/params-local.php
generate console/config/main-local.php
generate console/config/params-local.php
generate frontend/config/main-local.php
generate frontend/config/params-local.php
generate frontend/web/index-test.php
generate frontend/web/index.php
generate yii
chmod 0777 backend/runtime
chmod 0777 backend/web/assets
chmod 0777 frontend/runtime
chmod 0777 frontend/web/assets
chmod 0755 yii

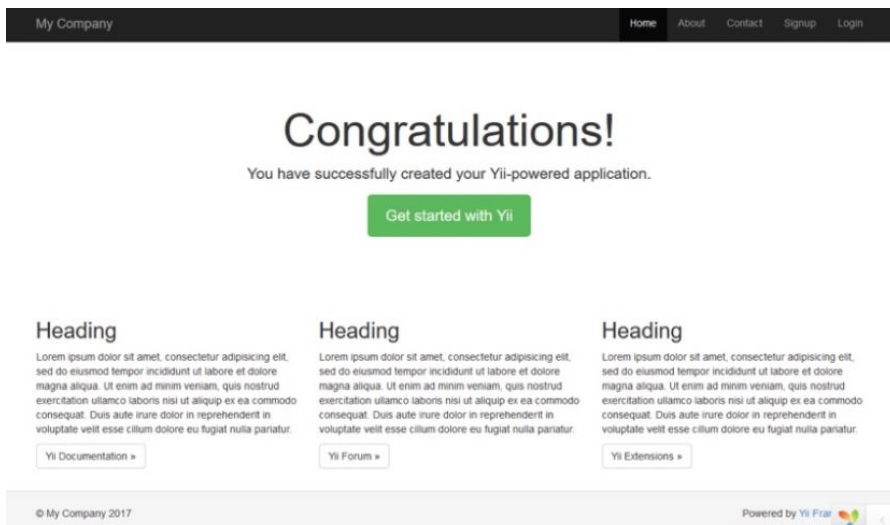
... initialization completed.

c:\xampp\htdocs\advanced>
```

selesai.JPG

**Gambar 2.16** Langkah Terakhir Install YII

8. Setelah selesai install akan masuk ke halaman ini

**Gambar 2.17** halaman YII

## 2.3 Penggunaan Database

### 2.3.1 Pengertian Database

Apa definisi atau arti dari basis data (database)? Basis data adalah sekumpulan data yang tersimpan secara sistematis di dalam komputer yang bisa diolah atau dimanipulasi dengan menggunakan perangkat lunak (program aplikasi) untuk menghasilkan informasi yang baru. Pendefinisian basis data meliputi spesifikasi berupa tipe data, struktur data dan juga batasan-batasan pada data yang kemudian disimpan.

Basis data merupakan aspek yang sangat penting dalam sistem informasi karena berfungsi sebagai gudang penyimpanan data yang akan diolah lebih lanjut. Basis data menjadi penting karena dapat mengorganisasi data, menghindari duplikasi data, menghindari hubungan antar data yang tidak jelas dan juga update yang rumit.

Proses untuk memasukkan dan juga mengambil data ke dan dari media penyimpanan data memerlukan perangkat lunak (software) yang disebut dengan sistem manajemen basis data (database management system — DBMS). DBMS merupakan sistem perangkat lunak yang memungkinkan pengguna basis data (database user) untuk memelihara, mengontrol dan mengakses data secara praktis dan efisien.

Dengan kata lain, semua akses ke basis data akan ditangani oleh DBMS. DBMS ini menjadi lapisan yang menghubungkan basis data dengan program aplikasi untuk memastikan bahwa basis data tetap terorganisasi secara konsisten dan dapat diakses dengan mudah.

Ada beberapa fungsi yang harus ditangani DBMS seperti pendefinisian data, menangani permintaan pengguna untuk mengakses data, memeriksa sekuriti dan integriti data yang didefinisikan oleh DBA (Database Administrator), menangani kegagalan dalam pengaksesan data yang disebabkan oleh kerusakan sistem maupun media penyimpanan (disk) dan juga menangani unjuk kerja semua fungsi secara efisien.

Tujuan utama DBMS adalah untuk memberikan tinjauan abstrak data kepada pengguna. Jadi sistem menyembunyikan informasi tentang bagaimana data disimpan, dipelihara dan juga bisa diakses secara efisien. Pertimbangan efisien di sini adalah rancangan struktur data yang kompleks tetapi masih bisa digunakan oleh pengguna awam tanpa mengetahui kompleksitas strukturnya.

### 2.3.2 Pembagian Database

Pembagian database Meliputi Menurut jenisnya, basis data dapat dibagi menjadi:

1. **Basis Data Flat-File.**

Basis data ini ideal untuk data berukuran kecil dan dapat dirubah dengan mudah. Pada dasarnya, basis data flat-file tersusun dari sekumpulan string dalam satu atau lebih file yang dapat diurai untuk mendapatkan informasi yang disimpan. Basis data flat-file cocok untuk menyimpan daftar atau data yang sederhana dan dalam jumlah kecil. Basis data flat-file akan menjadi sangat rumit apabila digunakan untuk menyimpan data dengan struktur kompleks walaupun dimungkinkan pula untuk itu.

Beberapa kendala dalam menggunakan basis data jenis ini adalah rentan pada korupsi data karena tidak adanya penguncian yang melekat ketika data digunakan atau dimodifikasi dan juga adanya duplikasi data yang mungkin sulit dihindari. Salah satu tipe basis data flat-file adalah file CSV yang menggunakan pemisah koma untuk setiap nilainya.

## 2. Basis Data Relasional.

Basis data ini mempunyai struktur yang lebih logis terkait cara penyimpanannya. Kata "relasional" berasal dari kenyataan bahwa tabel-tabel yang ada di basis data relasional dihubungkan satu dengan lainnya. Basis data relasional menggunakan sekumpulan tabel dua dimensi yang masing-masing tabel tersusun atas baris (tupel) dan kolom (atribut).

Untuk membuat hubungan antara dua atau lebih tabel, digunakan key (atribut kunci) yaitu primary key di salah satu tabel dan foreign key di tabel yang lain. Saat ini, basis data relasional menjadi pilihan utama karena keunggulannya. Program aplikasi untuk mengakses basis data relasional menjadi lebih mudah dibuat dan dikembangkan dibandingkan dengan penggunaan basis data flat-file.

Beberapa kekurangan yang mungkin dirasakan di basis data jenis ini adalah implementasi yang lebih sulit untuk data dalam jumlah besar dengan tingkat kompleksitasnya yang tinggi. Selain itu, proses pencarian informasi juga menjadi lebih lambat karena perlu menghubungkan tabel-tabel terlebih dahulu apabila datanya tersebar di beberapa tabel.

Namun, terlepas dari beberapa kekurangannya, basis data relasional telah digunakan secara luas. Saat ini, basis data relasional telah banyak dimanfaatkan oleh perusahaan-perusahaan dari skala kecil, menengah hingga besar. Beberapa basis data ternama yang ada saat ini, baik yang berasal dari sumber terbuka (open source) atau yang komersil, adalah juga basis data relasional.

### 2.3.3 Contoh Beberapa Basis Data

Berikut ini adalah beberapa basis data relasional populer dan paling banyak digunakan untuk saat ini beserta sejarahnya:

#### 1. Basis Data MySQL.



**Gambar 2.18** Basis Data MySQL

pada buku ini penulis menggunakan data base MySQL. MySQL merupakan basis data sumber terbuka yang paling populer dan banyak digunakan untuk aplikasi berbasis web seperti website dinamis dan e-commerce. Tahun 2013, MySQL merupakan basis data kedua yang paling banyak digunakan di dunia dan yang pertama untuk basis data sumber terbuka.

Dilihat dari sejarahnya, MySQL dibuat tahun 1995 dan disponsori oleh perusahaan Swedia, MySQL AB. Pengembang platform MySQL adalah Michael Widenius, David Axmark dan Allan Larsson. MySQL dibuat untuk menyediakan opsi pengelolaan data yang efisien, terpercaya dan handal. Pada tahun 2000, platform MySQL berubah menjadi sumber terbuka dan mengikuti ketentuan GPL.

Penggunaan MySQL sebagai basis data utama untuk aplikasi web sering dipadukan dengan PHP sebagai bahasa skrip berorientasi obyek. MySQL adalah salah satu komponen penting dari web service solution stack LAMP (Linux, Apache, MySQL and PHP) yaitu platform pengembangan web sumber terbuka dimana Linux sebagai sistem operasi, Apache sebagai Web Server, MySQL sebagai basis data dan PHP sebagai bahasa skrip.

Apabila Anda membuat blog atau website menggunakan CMS seperti Joomla, Wordpress, Drupal atau Magento, Anda sedang menggunakan MySQL sebagai solusi basis datanya. MySQL juga banyak digunakan oleh perusahaan-perusahaan besar dunia seperti Facebook, Google, Adobe, Alcatel Lucent dan juga Zappos.

Pada Januari 2008, MySQL diakuisisi oleh Sun Microsystems. Pada April 2009, terjadi pencapaian kesepakatan antara Sun Microsystems dan Oracle Corporation terkait pembelian Sun Microsystems beserta hak cipta (copyright) dan

merek dagang (trademark) MySQL oleh Oracle. Namun baru pada Januari 2010, MySQL secara resmi diakuisisi oleh Oracle.

Di bawah naungan Oracle Corporation, MySQL tersedia melalui skema lisensi ganda. Anda dapat menggunakan opsi lisensi sumber terbuka (GPL) selama masih mematuhi aturan lisensi tersebut. Jika Anda ingin mendistribusikan aplikasi non-GPL dimana terdapat MySQL di dalamnya, Anda dapat membeli lisensi komersial sebagai gantinya.

Berikut ini adalah beberapa keunggulan dari MySQL jika dibandingkan dengan pesaingnya:

- (a) Performa. Tidak dapat dipungkiri lagi bahwa MySQL lebih cepat dari RDBMS pesaingnya. Hasil perbandingan lima database sebagai database untuk aplikasi web, MySQL berada di urutan pertama kemudian diikuti oleh Oracle DB.
- (b) Biaya rendah. MySQL tersedia secara gratis (free) dibawah lisensi sumber terbuka (GPL) atau opsi dengan biaya rendah untuk lisensi komersialnya.
- (c) Mudah digunakan. Kebanyakan pengelolaan database modern saat ini menggunakan SQL sebagai bahasa standar. Namun apabila dibandingkan dengan RDBMS lainnya, MySQL masih lebih mudah dalam pengaturannya.
- (d) Portabilitas. MySQL dapat diindtal dan berjalan di semua platform yang ada saat ini seperti Windows, Linux dan juga Unix.
- (e) Kode sumber terbuka. Kode sumber MySQL dapat diperoleh secara gratis (free) dan juga bisa dimodifikasi sekalipun untuk kebanyakan penggunaanya ini tidak perlu melakukan.
- (f) Ketersediaan dukungan. Tidak semua produk sumber terbuka (open source) memiliki perusahaan induk yang dapat menawarkan dukungan, pelatihan dan sertifikasi, namun Anda bisa mendapatkan semua itu dari MySQL AB.

adapun kekurangan dari MySQL sebagai berikut :

(a) **Technical Support Kurang Baik**

Dibalik kelebihan dari MySQL, ada juga kekurangan dibaliknya, termasuk technical Support pada MySQL diklaim kurang bagus. Ini berhubungan dengan statusnya yang open source. Hal tersebut membuat user mengalami kesulitan ketika menghubungi technical support untuk mencari MySQL ketika dihadapkan pada suatu permasalahan pada saat menggunakan software tersebut.

(b) **Sulit Diaplikasikan dengan Database yang Besar**

MySQL memang tidak cocok untuk diterapkan pada perusahaan ataupun instansi yang besar. Karena mereka pasti menggunakan database dan jumlah data yang sangat besar. Itu dikarenakan MySQL merupakan DBMS

yang sangat ramah dengan spesifikasi komputer yang rendah, maka dari itu MySQL tidak memiliki fitur yang selengkap Oracle.

### (c) Tidak Populer Untuk Aplikasi Game dan Mobile

Jika kamu ingin membangun sebuah aplikasi mobile ataupun game, MySQL bukan salah satu database yang tepat untuk kamu gunakan. Karena DBMS ini sangat kurang digunakan untuk aplikasi-aplikasi tersebut.

## 2. Basis Data Oracle.



**Gambar 2.19** Basis Data Oracle

Oracle merupakan basis data relasional terkemuka yang dimiliki oleh Oracle Corporation. Oracle telah dianggap sebagai basis data terbaik untuk versi basis data komersial. Oracle sendiri tersedia dalam berbagai konfigurasi dengan cakupan tool yang dapat disesuaikan untuk perusahaan skala kecil, menengah hingga besar yang membutuhkan solusi yang terbaik dan tepat dari sebuah basis data untuk keperluan bisnisnya. Oracle dianggap lebih baik untuk masalah kinerja dan skalabilitas dibandingkan dengan basis data komersial lainnya.

Sejarah Oracle dimulai untuk pertama kali pada tahun 1977 ketika versi pertamanya dikembangkan oleh SDL (Software Development Laboratories) yang di dalamnya terdapat Larry Ellison dan dua orang temannya, Bob Miner dan Ed Oates. Nama Oracle sendiri berasal dari nama kode (code-name) sebuah proyek yang didanai oleh CIA ketika Larry Ellison masih bekerja di pekerjaan sebelumnya di Ampex.

Pada tahun 1979, versi komersial Oracle tersedia untuk pertama kali, sedangkan versi terkini Oracle adalah Oracle 12c, dimana "c" mengacu pada cloud computing (komputasi awan). Cloud computing merupakan refleksi kerja Oracle



untuk memperluas basis datanya yang memungkinkan perusahaan mengkonsolidasi dan mengelola basis data sebagai cloud service.

### **Kelebihan Database Oracle**

Tidak heran, Oracle merupakan salah satu vendor terdepan dalam pengaplikasian dan juga implementasi dari sistem database. Hal ini disebabkan karena Database Oracle memiliki banyak kelebihan. Berikut ini adalah beberapa kelebihan yang dimiliki oleh Database Oracle :

**(a) Memiliki kemampuan yang baik untuk melakukan manajemen sistem database**

Kelebihan Database Oracle yang pertama adalah mampu untuk melakukan manajemen sistem data dengan baik. Hal ini memungkinkan user ataupun programmer yang menggunakan Database Oracle dapat melakukan manajemen basis data dengan baik dan juga sempurna.

**(b) Jumlah data dan juga angka yang dihandle sangat besar**

Pengolahan data yang dapat dilakukan oleh Database Oracle merupakan salah satu pengolahan data yang cepat dan juga akurat. Yang penting adalah, Anda sebagai user ataupun programmer yang menggunakan Database Oracle mengerti cara mengoperasikannya, maka hasil dari pengolahan data akan muncul secara akurat tanpa ada kesalahan. Selain itu, berkat spesifikasi yang tinggi dari Oracle, sistem manajemen basis data ini juga dapat bekerja lebih cepat dibandingkan sistem DBMS yang lainnya.

**(c) Dapat mengolah data dengan cepat dan akurat**

Pengolahan data yang dapat dilakukan oleh Database Oracle merupakan salah satu pengolahan data yang cepat dan juga akurat. Yang penting adalah, Anda sebagai user ataupun programmer yang menggunakan Database Oracle mengerti cara mengoperasikannya, maka hasil dari pengolahan data akan muncul secara akurat tanpa ada kesalahan. Selain itu, berkat spesifikasi yang tinggi dari Oracle, sistem manajemen basis data ini juga dapat bekerja lebih cepat dibandingkan sistem DBMS yang lainnya.

**(d) Memiliki kemampuan untuk melakukan cluster server**

Merupakan keunggulan dari Database Oracle, yang mungkin tidak dimiliki oleh vendor DBMS lainnya. Database Oracle mampu untuk mengimplementasikan manajemen basis data yang bersumber dari multi server, atau yang dalam istilah Database Oracle disebut sebagai Technology Cluster Server. Hal ini memungkinkan Database Oracle dapat melakukan manajemen data dengan menggunakan lebih dari 1 server. Apabila suatu instansi

memiliki 50 server, maka Database Oracle mampu untuk mengolah seluruh data yang dimiliki oleh ke-50 server tersebut secara aktif, tanpa ada gangguan sama sekali.

**(e) Dapat melakukan management User**

Database Oracle juga memiliki kelebihan lainnya, yaitu mampu untuk melakukan management user. Hal ini berkaitan dengan pembatasan hak yang dimiliki oleh user, jadi Anda dapat melakukan hal tersebut dengan menggunakan Database Oracle.

**(f) Multi-Platform**

Multi platform, berarti Database Oracle bisa digunakan pada berbagai macam platform. Berbeda dengan Microsoft SQL Server yang hanya bisa digunakan pada Sistem Operasi Windows saja, maka Database Oracle bisa beroperasi pada Sistem Operasi lainnya, seperti Mac, dan juga Linux.

**(g) Pemrosesan data yang cepat**

Keunggulan utama dari sistem basis data yang diberikan oleh Database Oracle adalah pemrosesan dari pengolahan data yang cepat. Hal ini akan membantu meningkatkan efisiensi waktu dari suatu instansi, dan membuat banyak biaya penyelenggaraan IT menjadi lebih kecil.

**(h) Memiliki kemampuan flashback**

Flashback merupakan kemampuan untuk melakukan restore ulang pada suatu database yang tidak sengaja mengalami error, seperti mengalami mati listrik. Ketika anda melakukan manajemen database, pekerjaan yang sudah Anda lakukan akan tersimpan otomatis, sehingga ketika mengalami masalah dengan komputer ataupun mengalami mati listrik, pekerjaan Anda bisa di-restore kembali dengan menggunakan fitur flashback ini.

## **Kekurangan Database Oracle**

Meskipun memiliki banyak kelebihan dibandingkan dengan software DBMS lainnya, namun ternyata Database Oracle juga memiliki beberapa kekurangan. Berikut ini adalah beberapa kekurangan dari Database Oracle :

**(a) Harga software DBMS tinggi dan mahal**

Kekurangan pertama dari Database Oracle adalah masalah harga jual. Apabila dibandingkan dengan software DBMS yang lainnya, Database Oracle merupakan software DBMS dengan nilai yang paling tinggi dan paling mahal. Hal ini tentu saja akan mubazir apabila anda tidak terlalu membutuhkan spesifikasi software DBMS yang tinggi, seperti Database Oracle.

(b) **Spesifikasi minimum hardware yang tinggi**

Karena memiliki fungsi yang sangat banyak, dan juga kemampuan yang sangat tinggi dalam melakukan manajemen dan pengolahan dari suatu database, maka hal ini berpengaruh terhadap spesifikasi minimum yang dibutuhkan oleh sebuah komputer dalam menjalankan software Database Oracle. Tidak semua PC kantor biasa bisa menjalankan Database Oracle dengan baik, karena membutuhkan spesifikasi PC yang cenderung tinggi.

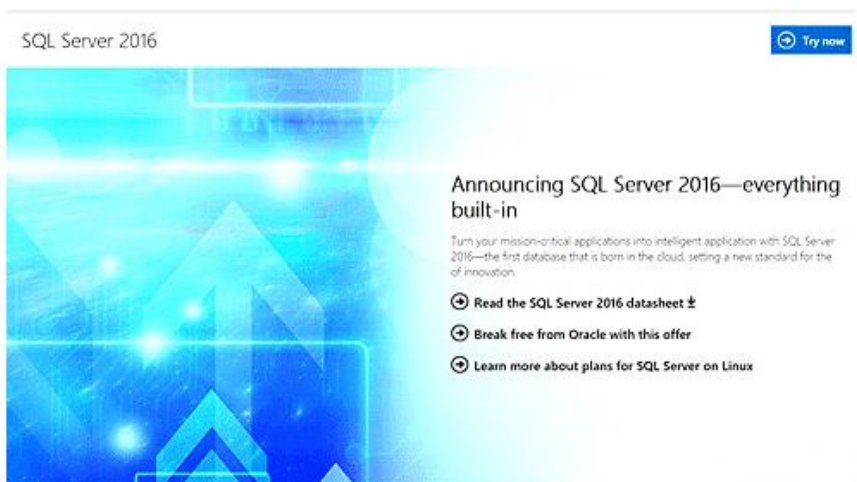
(c) **Tidak cocok untuk diaplikasikan pada perusahaan atau instansi kecil**

Harga yang mahal dan juga spesifikasi tinggi. Tentu saja ini membuat Database Oracle menjadi sangat tidak cocok untuk digunakan oleh perusahaan atau instansi kecil dan juga menengah. Spesifikasi dan juga fitur yang dimiliki oleh Database Oracle terlalu tinggi untuk diimplementasikan pada database instansi atau perusahaan kecil dan menengah.

(d) **Rumit dalam mengaplikasikannya**

Database Oracle ternyata juga mengalami kekurangan, terutama dalam hal pengoperasiannya. Untuk dapat mengaplikasikan Database Oracle, seorang user ataupun programmer harus benar-benar bisa memahami cara kerja dari Database Oracle. Hal ini disebabkan karena fitur yang tinggi, membuat Database Oracle juga membutuhkan operator yang berpengalaman dalam melakukan manajemen database.

### 3. Basis data Microsoft SQL Server.



**Gambar 2.20** Basis Data Microsoft SQL Server

Dari namanya, Anda bisa menebak dengan mudah siapa pemilik basis data ini. Benar, Microsoft tidak hanya terkenal dengan perangkat lunak sistem operasi Microsoft Windows dan Microsoft Windows Server, aplikasi perkantoran Microsoft Office dan programming language suite Microsoft Visual Studio, tetapi juga dengan basis datanya, Microsoft SQL Server.

Microsoft SQL Server adalah basis data relasional yang bersifat komersial. Tidak seperti Microsoft Office Access yang peruntukannya untuk komputer dekstop, Microsoft SQL Server untuk komputer dengan Windows Server yang menyediakan layanan pengelolaan basis data kelas perusahaan dan juga alat intelijen bisnis terpadu (integrated business intelligence (BI) tools).

Cukup banyak perusahaan skala kecil, menengah maupun besar dalam mengelola sistem informasi perusahaan menggunakan sistem operasi Microsoft Windows Server. Microsoft SQL Server tentu menjadi pilihan bagi pengguna Microsoft Windows Server karena selain mudah digunakan dan faktor ketersedian, juga telah terintegrasi dengan Microsoft Windows Server.

Versi terbaru basis data ini yaitu Microsoft SQL Server 2016 tersedia untuk platform on-premises (yang dikelola di server lokal) dan juga sebagai cloud database. Versi terbaru ini juga membawa fitur-fitur baru dan sejumlah peningkatan yang dapat memberi terobosan kinerja, keamanan lanjutan serta pelaporan dan analisis terpadu.

Awal sejarah Microsoft SQL Server dimulai ketika Microsoft dan Sybase mengembangkan sistem manajemen basis data yang diberi nama Sybase SQL Server untuk bisa berkompetisi dengan basis data yang sudah ada saat itu seperti IBM dan Oracle. Sybase akan mendapatkan hak penjualan dan keuntungan dari versi produk basis data yang tidak didesain untuk platform Microsoft. Pada tahun 1989, basis data pertama dirilis. Kemudian Sybase menjual hak ciptanya tersebut ke Microsoft dan akhirnya Microsoft memberi nama ulang basis data tersebut menjadi Microsoft SQL Server.

## **Kelebihan Microsoft SQL Server**

Saat ini SQL server sudah dikembangkan dan juga diluncurkan lebih dari 7 generasi, yang sangat mudah untuk digunakan terutama pada komputer dengan sistem operasi windows, baik 32 maupun 64 bit. Seperti kebanyakan vendor dan juga software dari DBMS, Microsoft SQL Server ini memiliki beberapa kelebihan. Berikut ini adalah beberapa kelebihan Microsoft SQL Server :

### **(a) Bekerja dengan sangat baik pada sistem operasi Windows**

Kelebihan dari Microsoft SQL Server yang pertama adalah tentu saja dapat bekerja dengan sangat baik pada semua sistem operasi windows, mulai dari Windows Xp, Vista, Seven, dan juga 8 berbagai versi. Hal ini tentu saja disebabkan karena Microsoft SQL Servers juga merupakan software DBMS

yang dibuat dan dikembangkan oleh Microsoft itu sendiri, maka dari itu Microsoft SQL Server sangat baik dan juga cocok apabila digunakan dan juga diaplikasikan pada sistem operasi windows.

**(b) Mendukung banyak software database**

Microsoft SQL Servers juga memiliki kelebihan lainnya, yaitu mendukung banyak sekali software database. Hal ini tentu saja akan membantu mempermudah pekerjaan anda sebagai seorang programmer untuk mengolah database. Anda dapat menggunakan software database apa pun yang ingin dan biasa anda gunakan, tanpa perlu khawatir tidak akan kompatibel dengan software Microsoft SQL Servers ini.

**(c) Dapat membuat clustering data**

Microsoft SQL Servers juga memiliki kelebihan lain yang cukup baik, yaitu dapat membantu melakukan pembuatan dari clustering data. Clustering data ini merupakan salah satu hal yang dapat membantu mempermudah pekerjaan dari manajemen basis data atau database, sehingga data yang akan diolah menggunakan software Microsoft SQL Servers ini akan menjadi lebih mudah untuk diimplementasikan, karena dapat terbagi menjadi beberapa cluster.

**(d) Pengendalian dari sebuah database yang terpusat**

Pengendalian dari sebuah database atau basis data dengan menggunakan software Microsoft SQL Servers ini juga akan menjadi terpusat. Hal ini berarti suatu database akan dikelola dengan baik dan juga dikelola secara terpusat, sehingga tidak dimungkinkan terjadinya suatu miskomunikasi karena kesalahan dalam mengolah dan mengimplementasikan database. Dengan hanya penggunaan satu komputer saja untuk melakukan pengolahan database, maka kesalahan pun dapat diminimalisir.

**(e) Dapat melakukan backup data**

Kelebihan yang terakhir dari Microsoft SQL Servers ini adalah dapat melakukan backup data. Data yang diimplementasikan ke dalam suatu sistem melalui software Microsoft SQL Servers ini dapat di backup, sehingga bisa digunakan kembali ketika akan mengimplementasikan sistem yang sama atau mirip, dan juga untuk menjaga agar ketika terjadi gangguan pada sistem bisa restore kembali berkat adanya fitur backup data ini.

**(f) Management password yang baik dan aman**

Keamanan dari Microsoft SQL Servers ini juga sangat baik. Hal ini akan berpengaruh terhadap ketahanan keamanan dari suatu database dan juga servernya. Menggunakan password yang akan sulit untuk dihack, maka

Microsoft SQL Servers akan membantu data yang ada pada database anda menjadi lebih aman dari tangan tangan yang tidak bertanggung jawab.

### **Kekurangan Microsoft SQL Server**

Selain beberapa kelebihan yang dimiliki oleh Microsoft SQL server, ternyata ada juga beberapa kekurangan yang dimiliki oleh SQL Server. Berikut ini adalah beberapa kekurangan dari Microsoft SQL Server :

**(a) Harga belinya yang lumayan tinggi**

Harga jual dari produk software Microsoft memang cenderung tinggi dan mahal apabila dibandingkan dengan software lainnya. Hal ini memang merupakan salah satu hal yang wajar, karena Microsoft SQL Servers merupakan salah satu software yang sangat kompeten di bidang DBMS.

**(b) Tidak multi OS, hanya bisa digunakan pada sistem operasi yang dibuat oleh Microsoft**

Kekurangan lainnya dari software Microsoft SQL Servers ini yang berikutnya adalah hanya identik dengan sistem operasi Windows saja. Bagi anda yang menggunakan sistem operasi Mac ataupun Linux dan Ubuntu, maka anda hanya bisa gigit jari, karena Microsoft SQL Server belum mendukung multi OS.

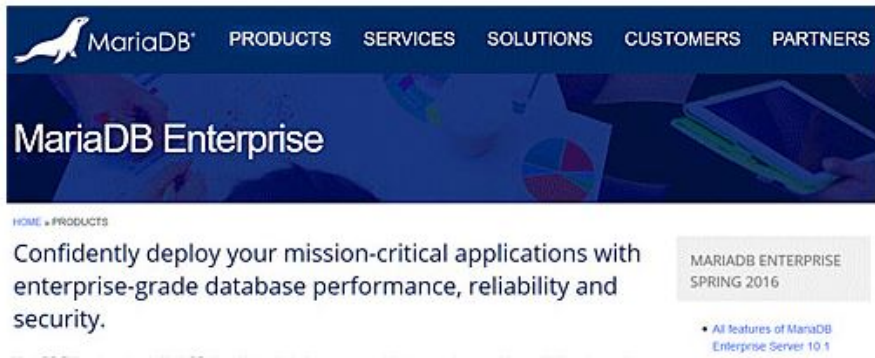
**(c) Sulit digunakan pada bahasa pemrograman lain**

SQL server terkenal dengan jodohnya bahasa pemrograman .net. Jika anda menggunakan bahasa pemrograman lain maka anda harus bersiap siap untuk bersusah payah karena bukan jodohnya. Selain itu DBMS ini juga membutuhkan fungsi CPU yang cukup besar.

**(d) Tidak cocok untuk skala database besar**

Tidak se bagus performa Oracle, jika digunakan pada sistem berskala besar, performance database kalah di bandingkan dengan Oracle. Jadi jika anda menjalankan sistem skala besar, kemungkinan kurang begitu cocok.

## **4. Basis Data MariaDB.**



**Gambar 2.21** Basis Data MariaDB

Mungkin sebagian dari Anda masih asing atau pernah mendengar nama basis data ini. MariaDB merupakan sistem basis data relational yang sepenuhnya sumber terbuka. Semua kode sumber basis data MariaDB dirilis di bawah lisensi GPL, LGPL atau BSD. MariaDB sebenarnya adalah fork dari basis data MySQL.

Fork di sini dapat diartikan sebagai proyek terkait yang dapat dianggap sebagai versi lain (mini) dari MySQL standar. MariaDB merupakan pengembangan versi lain MySQL oleh Michael Widenius yang telah meninggalkan Sun Microsystems setelah diakuisisi secara resmi oleh Oracle Corporation.

Kepopuleran basis data ini terus merangkak naik. Ini terbukti pada tahun 2013, Red Hat Enterprise Linux (RHEL) telah menyingkirkan MySQL dan kemudian menggantinya dengan MariaDB. Demikian juga dengan Fedora yang lebih memilih MariaDB dibandingkan MySQL untuk rilis Fedora 19.

Rilis stabil terbaru MariaDB adalah versi 10.1 yang juga dikenal dengan MariaDB Enterprise 2016. Di rilis stabil terbarunya ini, MariaDB memiliki proteksi yang lebih baik dengan pengamanan basis data pada setiap lapisan. MariaDB dapat mendeteksi dan mencegah serangan dengan menggunakan access control, Key Management Services dan juga authentication.

## **Kelebihan MariaDB**

Adapun kelebihan MariaDB sebagai berikut:

### **(a) Portabilitas**

Kelebihan pertama yakni aplikasi ini masuk ke dalam kategori aplikasi portabel. Mengapa? Karena mampu dipasang dan berjalan pada semua sistem operasi yang ada. Jadi, bagi anda yang menggunakan sistem operasi

seperti Linux, Windows, Mac Os X, FreeBSD, dan lain sebagainya tidak perlu khawatir apakah aplikasi ini cocok diinstal ke sistem operasi tersebut.

(b) **Open source**

kedua yakni MariaDB merupakan aplikasi yang bersifat open source. Kami yakin anda sudah paham tentang maksud istilah ini. Intinya, anda bisa menggunakan MariaDB secara gratis, termasuk juga menggunakan seluruh fitur yang ada. Namun yang harus anda perhatikan yakni anda tidak boleh menggandakan atau malah menjual aplikasi ini dengan tujuan komersil.

(c) **Multi-user**

Seperti yang sudah kami terangkan bahwa MariaDB mampu digunakan oleh banyak user atau pengguna dalam waktu yang bersamaan. Ketika digunakan secara bersama-sama, aplikasi ini tidak akan membuat anda frustasi karena lemot. Justru software ini terkenal stabil dan akan membuat anda nyaman menggunakannya.

(d) **Aplikasi database yang cepat**

Kelebihan MariaDB selanjutnya adalah berkaitan dengan performance tuning. Aplikasi ini mampu berjalan dalam waktu yang lumayan cepat ketika ingin menangani query yang sederhana. Artinya, data akan lebih banyak memproses SQL untuk satu waktu.

(e) **Memiliki banyak pilihan data**

juga merupakan aplikasi yang telah menyediakan banyak pilihan data yang bisa memberikan banyak alternatif untuk pengguna. Beberapa pilihan data yang disediakan antara lain float, double, signed or unsigned integer, date, text, char, timestamp, dan lain sebagainya.

(f) **Keamanan**

Kita tahu bahwa keamanan merupakan bagian yang paling penting yang menjadi pertimbangan ketika kita ingin menggunakan sebuah aplikasi. MariaDB sudah dilengkapi dengan beberapa firewall misalnya level subnetmask, izin akses, nama host, dan berbagai perizinan lain yang bersifat detail yang harus dilalui semua itu untuk bisa masuk ke dalam sistem.

(g) **Mampu digunakan untuk skala besar**

Kelebihan MariaDB dibandingkan software lain adalah aplikasi ini dapat digunakan dalam skala yang besar. Inilah mengapa MariaDB lebih cocok digunakan untuk organisasi atau perusahaan yang memang memiliki database dalam jumlah yang besar.

(h) **Dilengkapi dengan 20 bahasa**



Dengan dilengkapi dengan bahasa sebanyak 20, MariaDB mampu mendeteksi pesan eror yang diterima oleh klien. Aplikasi ini akan langsung mendeteksi kesalahan meskipun sebenarnya anda merasa kesalahan pesan tersebut tidak terlalu besar.

### Kekurangan MariaDB

Lalu apa saja hal-hal yang menjadi kekurangan dari MariaDB? Simak jawaban di bawah ini:

1. MariaDB merupakan aplikasi yang hanya disarankan untuk digunakan oleh perusahaan besar. Jadi, untuk perusahaan kecil disarankan menggunakan aplikasi yang lain.
2. Konon MariaDB memiliki keterbatasan di dalam menyimpan data pada saat data yang disimpan tersebut sudah berada di ambang batas atau bahkan melebihi kapasitas yang mampu ditampung oleh server.
3. Ternyata bahasa Indonesia tidak termasuk ke dalam salah satu diantara 20 bahasa yang akan digunakan untuk menemukan pesan eror.

## 2.4 Penggunaan XAMPP

### 2.4.1 Pengertian XAMPP

XAMPP adalah sebuah paket perangkat lunak (software) komputer yang sistem penamaannya diambil dari akronim kata Apache, MySQL (dulu) / MariaDB (sekarang), 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 OS Linux, OS Windows, Mac OS, dan juga Solaris.

Sejarah mencatat, software XAMPP pertama kali dikembangkan oleh tim proyek bernama Apache Friends dan sampai saat ini sudah masuk dalam rilis versi 7.3.9 yang bisa didapatkan secara gratis dengan label GNU (General Public License).

Jika dijabarkan secara gamblang, masing-masing huruf yang ada di dalam nama XAMPP memiliki arti sebagai berikut ini: **X = Cross Platform**

Merupakan kode penanda untuk software cross platform atau yang bisa berjalan di banyak sistem operasi. **A = Apache**

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

Sementara itu, untuk huruf P selanjutnya merupakan singkatan dari bahasa pemrograman Perl yang kerap digunakan untuk memenuhi berbagai macam kebutuhan. Perl ini bisa berjalan di dalam banyak sistem operasi sehingga sangat fleksibel dan banyak digunakan.

## 2.4.2 Fungsi XAMPP

Program aplikasi XAMPP berfungsi sebagai server lokal untuk mengampu berbagai jenis data website yang sedang dalam proses pengembangan. Dalam prakteknya, XAMPP bisa digunakan untuk menguji kinerja fitur ataupun menampilkan konten yang ada didalam website kepada orang lain tanpa harus terkoneksi dengan internet, atau istilahnya website offline. XAMPP bekerja secara offline layaknya web hosting biasa namun tidak bisa diakses oleh banyak orang. Maka dari itu, XAMPP biasanya banyak digunakan oleh para mahasiswa maupun pelajar untuk melihat hasil desain website sebelum akhirnya dibuat online menggunakan web hosting yang biasa dijual dipasaran.

## 2.4.3 Komponen Penting di Dalam XAMPP

### 1. Htdocs

Htdocs merupakan nama sebuah folder bagian dari XAMPP yang berfungsi untuk menyimpan berbagai file dan dokumen yang akan ditampilkan kedalam website. Kapasitas penyimpanan di dalam menu htdocs sendiri bergantung kepada harddisk yang terpasang di perangkat komputer.

### 2. Control Panel

Seperti namanya, Control Panel memberikan kita akses lebih leluasa dalam mengatur database, mengunggah file, atau melakukan setting lebih detail terkait jeroan website. Dari dalam Control Panel kita juga bisa mengatur stop dan star aplikasi XAMPP agar tidak memberatkan kinerja komputer.

### 3. PhpMyAdmin

Sama seperti fungsi PhpMyAdmin pada web hosting sesungguhnya, peran PhpMyAdmin dalam aplikasi XAMPP juga diplot sebagai pengatur konfigurasi MySQL. Untuk membuka phpMyAdmin di XAMPP caranya Anda bisa langsung mengetikkan URL <http://localhost/phpMyAdmin>

### 2.4.4 Cara MengInstall XAMPP

Setelah mengenal lebih jauh tentang pengertian XAMPP, Anda tentu penasaran dong bagaimana langkah-langkah cara menginstall XAMPP di Laptop atau PC? Caranya tidak jauh berbeda kok dibandingkan dengan cara menginstall aplikasi komputer pada umumnya. Berikut panduan instalasi XAMPP selengkapnya:

1. langkah pertama harus menyiapkan file XAMPP terlebih dahulu dengan cara mengunduh lewat link berikut ini <http://www.apachefriends.org/en/index.html>.. Tenang, file ini bersifat gratis , jadi Anda tidak perlu mengeluarkan biaya lisensi sepeserpun.

Apache Friends   Download   Add-ons   Hosting   Community   About   Search...   EN

## Download

XAMPP is an easy to install Apache distribution containing MariaDB, PHP, and Perl. Just download and start the installer. It's that easy.

Version	Checksum	Size
7.2.26 / PHP 7.2.26	What's Included? <b>md5 sha1</b>	<a href="#">Download (64 bit)</a> 145 Mb
7.3.13 / PHP 7.3.13	What's Included? <b>md5 sha1</b>	<a href="#">Download (64 bit)</a> 146 Mb
7.4.1 / PHP 7.4.1	What's Included? <b>md5 sha1</b>	<a href="#">Download (64 bit)</a> 146 Mb

[Requirements](#)   [Add-ons](#)   [More Downloads »](#)

Documentation/FAQs

There is no real manual or handbook for XAMPP. We wrote the documentation in the form of FAQs. Have a burning question that's not answered here? Try the Forums or Stack Overflow.

- Linux FAQs
- Windows FAQs
- OS X FAQs
- OS X XAMPP-VM FAQs

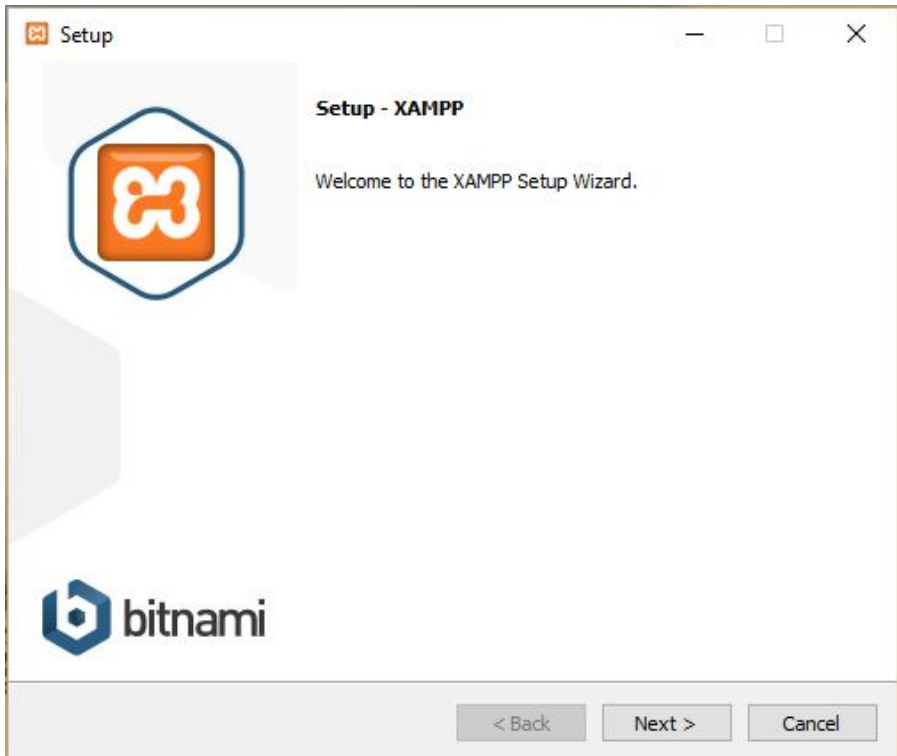
Add-ons and Themes

WordPress   Joomla   Drupal   Magento

sefriends.org/index.html dows XP or 2003 are not supported. You can download a compatible version of XAMPP for these

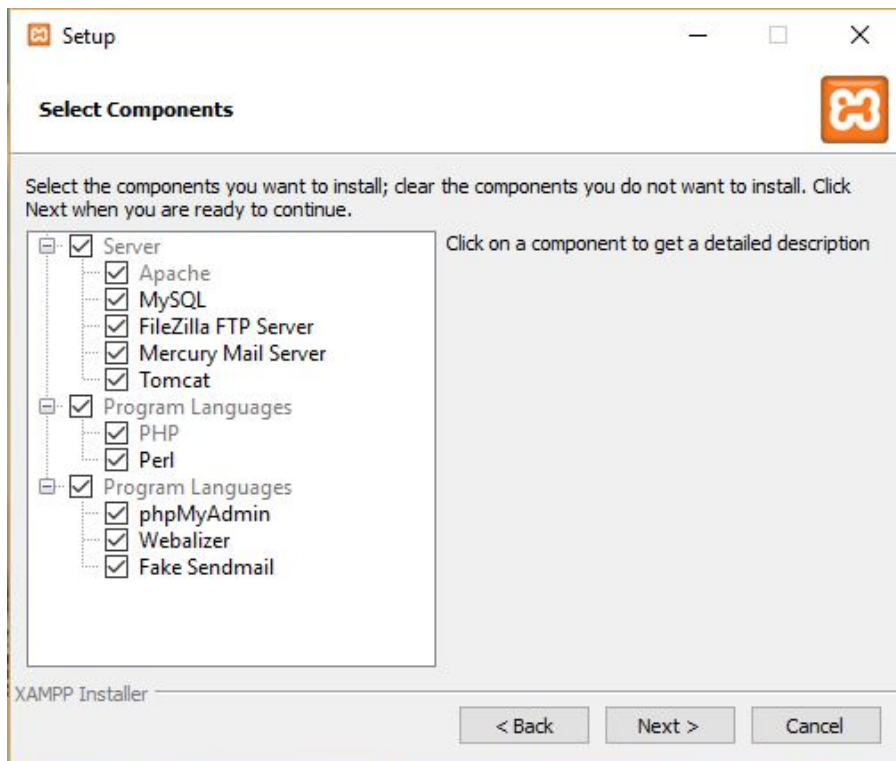
**Gambar 2.22**    Halaman Download Web Aplikasi Xampp

2. Setelah file berhasil didownload dengan sempurna langkah selanjutnya adalah menyiapkan space/ruang harddisk di laptop atau komputer dengan minimal masih mnyisakan ROM sebesar 1GB guna menampung gambar, teks, video, dan dokumen website.
3. Jalankan file **xampp-windows-x64-7.3.9-0-VC15-installer** atau versi lainnya yang lebih mutakhir.
4. Akan ada jendela baru yang terbuka sebagai indikasi dimulainya proses instalasi. Pada proses ini Anda akan diminta untuk memilih bahasa Indonesia atau English, pilih saja yang bahasa Indonesia kemudian klik next.



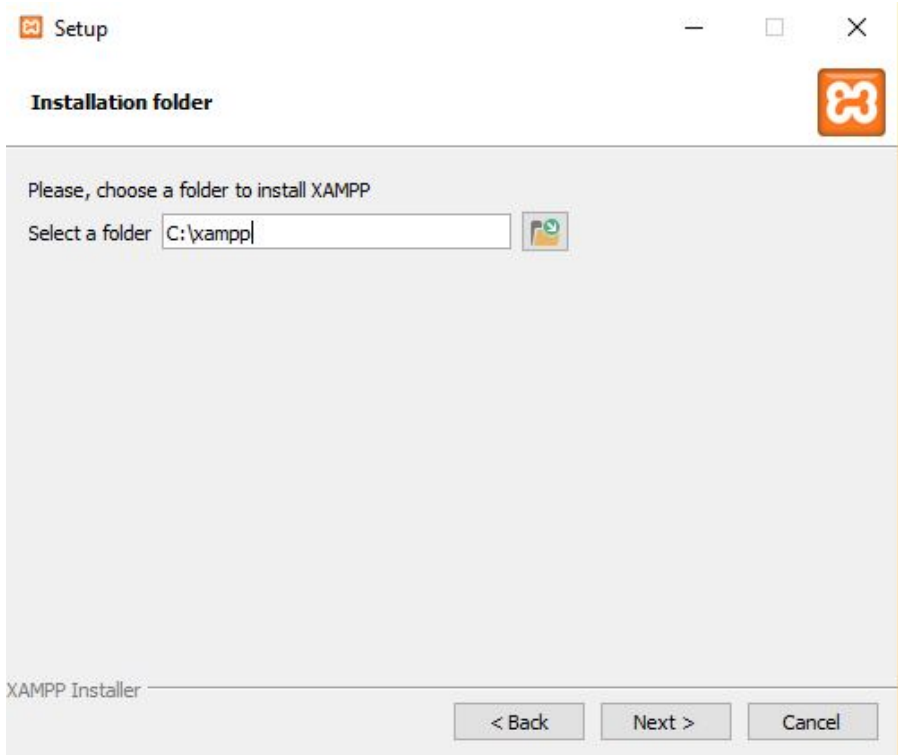
**Gambar 2.23** langkah pertama install xampp

- Setelah itu, Anda akan diminta untuk memilih komponen yang akan digunakan. Standar untuk server web berbasis CMS WordPress menggunakan MySQL, FileZilla FTP server, dan Apache, sedangkan dari bahasa pemrograman bisa menceklist pilihan phpMyAdmin, PHP, dan Perl. Kemudian klik next.



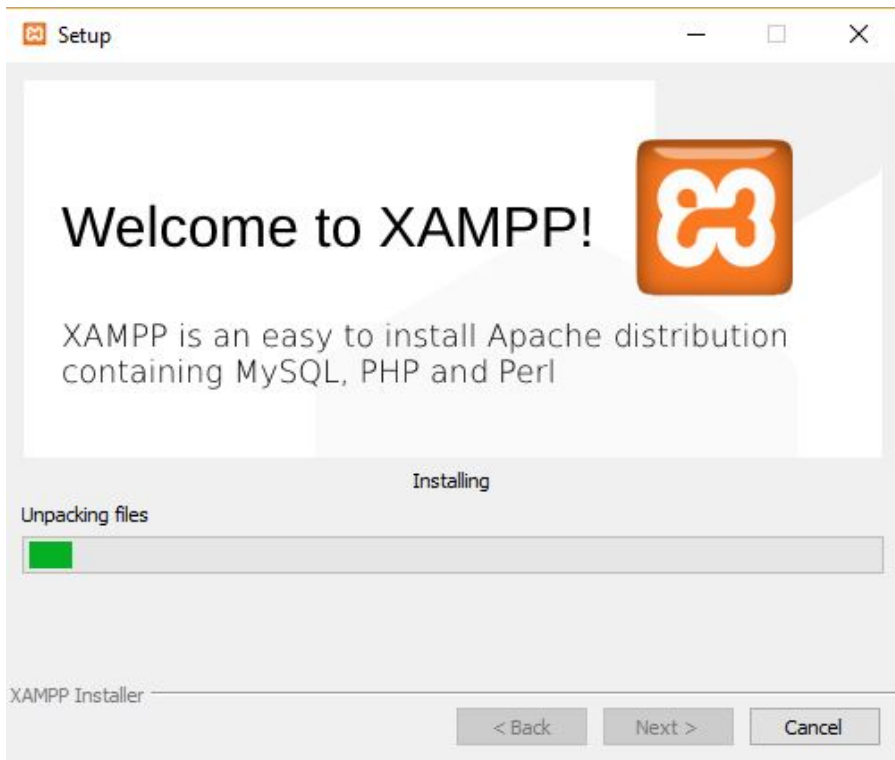
**Gambar 2.24** langkah install xampp

6. Pada tahap ini kita diminta untuk memilih lokasi file XAMPP yang akan diinstal, gunakan saja pengaturan default lalu klik next/install.



**Gambar 2.25** menentukan lokasi file disimpan

7. Tunggu beberapa menit sampai proses instalasi XAMPP selesai yang ditandai dengan bar progress seluruhnya berubah menjadi warna hijau.



**Gambar 2.26** proses install XAMPP

8. Anda sudah bisa menjalankan program XAMPP di laptop atau PC Windows untuk membuat website secara offline menggunakan localhost.

## 2.5 Penggunaan Aplikasi *Text Editor*

Text editor terbaik seperti peralatan tukang yang lengkap. Jika peralatan tukang untuk membangun gedung, text editor berguna untuk membangun aplikasi. Text editor berguna berguna untuk menyusun baris kode agar menjadi aplikasi yang andal.

Pekerja bangunan jika peralatannya tidak lengkap tentu saja hasilnya tidak maksimal. Sama halnya dengan seorang pengembang website (developer dan programmer). Jika menggunakan text editor yang tidak lengkap atau tidak sesuai dengan kebutuhan, tentu saja hasilnya bisa kurang maksimal minimal kesusahan di dalam proses pengembangan.

Nah! Supaya Anda bisa memilih text editor yang sesuai dengan kebutuhan, pada artikel ini akan membahas apa itu text editor dan beberapa contoh text editor terbaik saat ini. Text Editor yang di terapkan pada buku ini yaitu Sublime Text 3 yang akan di jelaskan juga di bawah ini.

### 2.5.1 Apa Itu Text Editor?

Text editor adalah sebuah aplikasi yang biasa dipakai programmer untuk menyusun kode program sehingga menjadi aplikasi baru. Bagi Anda pengguna Windows mungkin pernah membuka aplikasi Notepad. Aplikasi tersebut merupakan contoh dari text editor yang ada saat ini.

Di bagian pembuka tadi sudah membahas sedikit mengenai pentingnya text editor. Ya, text editor merupakan bagian yang cukup penting bagi programmer. Bahkan bisa dibilang text editor merupakan senjata utama para programmer untuk membangun sebuah aplikasi. Apalagi coding (menyusun baris kode) merupakan bagian yang tidak pernah lepas dari pekerjaan seorang programmer.

Ada banyak sekali pilihan text editor. Mulai dari text editor dengan fitur yang sederhana sampai dengan text editor dengan fitur yang sangat lengkap dengan berbagai plugin tambahan yang tersedia. Namun, Anda tidak bisa memilih text editor berdasarkan fitur saja.

Setidaknya ada tiga faktor yang perlu Anda perhatikan ketika memilih text editor seperti kebutuhan fitur, spesifikasi komputer yang digunakan, dan bahasa pemrograman yang dipakai. Contohnya ketika Anda sedang mengembangkan aplikasi menggunakan dengan C akan lebih kompatibel jika memakai Visual Studio. Namun kembali lagi, itu juga tergantung kenyamanan masing-masing pengguna text editor.

Jika Anda masih bingung menentukan text editor yang sesuai dengan kebutuhan. Di bawah ini beberapa daftar text editor terbaik saat ini.

### 2.5.2 Macam-macam Text Editor

Ada berbagai macam text editor yang populer saat ini. Di antaranya mempunyai kelebihan fitur yang lengkap, ringan, atau dukungan dengan berbagai bahasa pemrograman maupun integrasi dengan program lain.

Pada bagian ini akan membahas text editor antara lain Atom dan Sublime Text yang mendukung di berbagai macam sistem operasi. Ada juga Vim yang termasuk text editor paling ringan di antara yang lainnya.

#### 1. Sublime Text



**Gambar 2.27** Logo Sublime



Sublime Text 3 adalah sebuah software yang dikembangkan oleh Jon Skinner. Beliau merupakan seorang programmer dari Australia.

Sublime text 3 merupakan aplikasi text editor untuk menulis kode. Banyak sejumlah bahasa program yang ada pada aplikasi ini. Diantaranya PHP, CSS, C, C++, HTML, ASP, Java, dan sebagainya. Tentu saja, software ini bisa lebih memudahkan pekerjaan pengguna saat membuat sebuah program.

Salah satu keunggulan dari Sublime Text memang tampilannya yang dinamis. Tersedia juga beberapa tema yang dapat Anda pilih sesuai dengan selera.

Sublime Text termasuk di dalam text editor yang mendukung berbagai macam sistem operasi seperti Windows, Mac, dan Linux. Jadi tidak masalah jika Anda mengerjakan pengembangan aplikasi menggunakan beberapa sistem operasi yang berbeda-beda.

Linux merupakan sistem operasi yang paling kompatibel dengan Sublime Text, performanya lebih maksimal di sistem operasi ini. Fungsi pencarian Sublime Text dapat bekerja lebih optimal di Linux karena terdapat

Keunggulan lain dari Sublime Text adalah cepat dan dapat digunakan bahkan di perangkat dengan resource yang kecil. Sublime Text juga memungkinkan Anda untuk menambah fitur dengan menginstall plugin. Anda bisa menginstall beberapa plugin apabila mempunyai resource yang cukup.

Sublime Text dapat Anda gunakan secara gratis dengan batasan waktu. Namun Sublime Text juga menyediakan fitur premium yang hanya bisa Anda gunakan jika melakukan pembayaran. Biayanya sekitar 70 US Dollar untuk mendapatkan lisensi penuh.

### **Keunggulan Sublime**

- (a) Keunggulan pertama adalah adanya multiple selection. Fungsinya yaitu untuk menghasilkan perubahan pada sebuah kode dalam waktu yang sama dan dalam baris yang berbeda. Multiple selection ini juga adalah salah satu fitur unggulan dari Sublime Text Editor. Anda bisa saja meletakkan kursor pada kode yang akan di ubah / edit, lalu tekan Ctrl + click atau blok kode yang akan diubah kemudian Ctrl + D. Nah, setelahnya Anda bisa merubah kode secara bersamaan.
- (b) Keunggulan kedua adalah adanya command pallete. Fitur itu mempunyai fungsi yang berguna untuk mengakses file shortcut dengan gampang. Untuk mencari file tersebut Anda bisa tekan Ctrl + Shift + P , kemudian cari perintah yang memang Anda inginkan.
- (c) Keunggulan ketiga adalah adanya distraction free mode. Fitur satu ini punya fungsi untuk merubah tampilan layar menjadi penuh dengan menekan SHIFT + F11. Fitur ini sebenarnya memang sangat dibutuhkan ketika Anda ingin fokus pada bagian pekerjaan tertentu yang sedang dikerjakan saja.
- (d) Keunggulan keempat adalah adanya find in project. Dalam fitur ini Anda bisa mencari dan membuka file di dalam sebuah project dengan cepat dan

mudah. Anda tinggal tekan Ctrl + P dan Anda sudah bisa mencari file yang diinginkan.

- (e) Keunggulan terakhir, plugin yang berbasis Python Plugin API. Teks editor ini punya plugin yang sangat beragam dan ini bisa memudahkan Anda dalam mengembangkan software-nya.

## 2. Visual Code Studio



**Gambar 2.28** logo Visual Code Studio

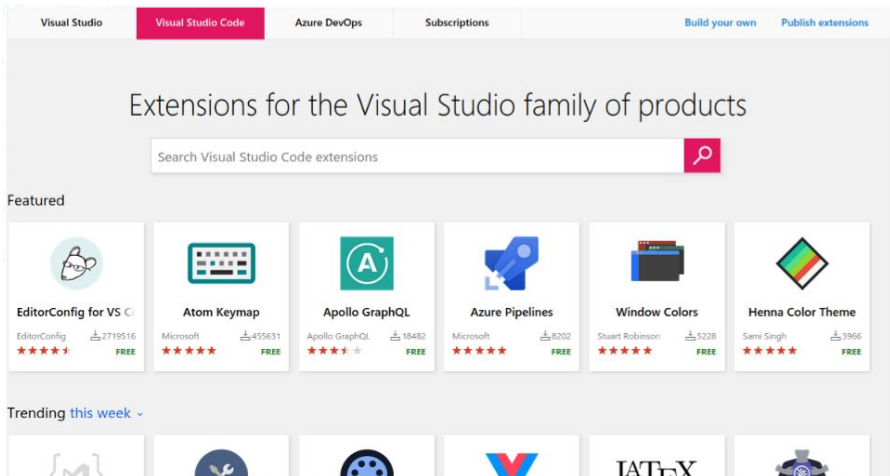
Visual Studio Code (VS Code) ini adalah sebuah teks editor ringan dan handal yang dibuat oleh Microsoft untuk sistem operasi multiplatform, artinya tersedia juga untuk versi Linux, Mac, dan Windows. Teks editor ini secara langsung mendukung bahasa pemrograman JavaScript, Typescript, dan Node.js, serta bahasa pemrograman lainnya dengan bantuan plugin yang dapat dipasang via marketplace Visual Studio Code (seperti C++, C, Python, Go, Java, dst).

Banyak sekali fitur-fitur yang disediakan oleh Visual Studio Code, diantaranya Intellisense, Git Integration, Debugging, dan fitur ekstensi yang menambah kemampuan teks editor. Fitur-fitur tersebut akan terus bertambah seiring dengan bertambahnya versi Visual Studio Code. Pembaruan versi Visual Studio Code ini juga dilakukan berkala setiap bulan, dan inilah yang membedakan VS Code dengan teks editor-teks editor yang lain.

Teks editor VS Code juga bersifat open source, yang mana kode sumbernya dapat kalian lihat dan kalian dapat berkontribusi untuk pengembangannya. Kode sumber dari VS Code ini pun dapat dilihat di link Github. Hal ini juga yang membuat VS Code menjadi favorit para pengembang aplikasi, karena para pengembang aplikasi bisa ikut serta dalam proses pengembangan VS Code ke depannya.

VS Code dapat digunakan secara langsung tanpa perlu ekstensi selama alur pemrogramannya sudah didukung langsung. Namun ada beberapa alur pemrograman yang tidak didukung langsung misalnya penggunaan snippets code dan

kemampuan Intellisense yang masih kurang. Disini letak kegunaan ekstensi karena dapat menambah kemampuan dukungan alur pemrograman yang diinginkan. Ekstensi VS Code ini banyak dibuat oleh pengembang-pengembang pihak ketiga yang juga menjadi kontributor dari VS Code. Ekstensi untuk VS Code dapat dilihat di Pasar Ekstensi dengan tautan berikut ini



**Gambar 2.29** Extensions

### 3. Notepad++



**Gambar 2.30** logo Notepad++

Notepad++ adalah suatu text editor yang berjalan pada Operating System(OS) Windows. Notepad++ disini menggunakan komponen-komponen Scintilla agar dapat menampilkan dan menyunting text dan berkas source code berbagai bahasa pemrograman. Notepad++ didistribusikan sebagai Free Software (gratis)

Proyek ini dilayani oleh Sourceforge.net dengan telah diunduh lebih dari 27 juta kali dan dua kali memenangkan penghargaan *SourceForge Community Choice Award for Best Developer Tool*.

Pengembang dari Notepad++ disini adan Don Ho yang dirilis pada tanggal 24 November 2003, dengan memiliki *license dari GNU General Public License* dengan ukuran program yang kecil yaitu 5.5MB. Bahasa pemrograman yang didukung oleh notepad++ adalah bahasa C++ karena fungsi-fungsinya yang dimasukkan kedalam daftar fungsi dan kata-katanya akan berubah sesuai dengan makna kata C++. Berikut ini adalah bahasa pemrograman yang didukung oleh Notepad++ sejak dirilis pada versi 5.9.3 :

- (a) ActionScript, Ada, ASP, Assembler, autoIt
- (b) Batch
- (c) C, C++, C, Caml, Cmake, COBOL, CSS
- (d) D, Diff
- (e) Flash ActionScript, Fortran
- (f) Gui4CLI
- (g) Haskell, HTML
- (h) Berkas INI, InnoSetup
- (i) Java, Javascript, JSP
- (j) KiXtart
- (k) LISP, Lua
- (l) Makefile, Matlab, MS-DOS
- (m) NSIS
- (n) Objective-C
- (o) Pascal, Perl, PHP, Postscript, PowerShell, Properties file, Python
- (p) R, Resource file, Ruby
- (q) Shell, Scheme, Smalltalk, SQL
- (r) TCL, TeX
- (s) Pascal, Perl, PHP, Postscript, PowerShell, Properties file, Python
- (t) XML
- (u) YAML

### **Keunggulan Notepad++ Dalam Software Web Programming**

- (a) ***Simple***, Ringan dan Cepat dibandingkan dengan text editor lainnya, notepad++ tidak perlu menunggu loading opening library, terlebih seperti pada software adobe dreamweaver dan eclipse apa lagi untuk PC / Laptop yang memiliki specification yang rendah.

- (b) **Bracket Matching** atau bisa dibilang mengumpulkan yang sesuai (berpasangan), biasanya digunakan pada saat menuliskan syntax percabangan, perulangan dan bagian utama program. Fungsi ini berguna untuk kita jika menuliskan block program yang panjang dan terkadang kita lupa untuk menutup block program yang telah kita buat
- (c) **Syntax Highlighting** tampilan source code, disini kita bisa melihat warna pada setiap fungsi dari syntax. Jadi kita tidak bingung menggolongkan kegunaan syntax yang kita tulis dan dapat dibaca dengan mudah. Contoh tulisan yang berwarna hijau biasanya terdapat pada statment jika kita menuliskan komentar pada sebuah program.
- (d) **Syntax Folding** atau melipat source code, ini hampir sama seperti bracket matching sebelumnya. Jika bracket matching digunakan untuk menunjukan awal dan akhir suatu block program. Syntax Folding disini tidak jauh beda tetapi, digunakan untuk menyembunyikan block program tertentu agar terlihat lebih ringkas tampilannya agar programmer tidak perlu melihat seluruh syntaxnya apa lagi jika sudah sampai lebih 1000 baris lebih.
- (e) **Quick Color Picker++** fungsi ini berguna pada saat kita menuliskan kode warna pada html atau pun CSS tetapi tidak harus menuliskan kodenya terlebih jika kita lupa kode pada warna yang kita inginkan, biasanya akan muncul kotak dengan banyak aneka warna yang kita tinggal pilih, setelah memilih nanti kode warna tersebut akan muncul.
- (f) **FingerText**, biasa digunakan untuk menuliskan bahasa PHP di notepad++, fungsi ini berfungsi untuk memudahkan pengetikan syntax dengan kata tertentu sebagai pemicu/trigger dan menggantikannya dengan menekan tombol TAB.

#### 4. Komodo Edit



**Gambar 2.31** logo Kodomo Edit

Komodo Edit adalah software editor program multi bahasa gratis yang dapat membuat penulisan kode program berkualitas menjadi mudah. Pengeditan program untuk bahasa dinamis termasuk Perl, PHP, Python, Ruby dan Tcl, ditambah dukungan kode browser sampling termasuk CSS, JavaScript, HTML, dan XHTML.

Pemeriksaan sintaks latar belakang dan penawaran sintaks untuk menangkap kesalahan dengan segera, sementara auto komplit dan calltips membimbing Anda ketika menulis. Tersedia untuk Windows, Linux, dan Mac OS X.

### **Editor Canggih**

- (a) Semua fundamental: Melipat kode, mengedit kolom / blok, mengomentari blok / baris, dan banyak lagi.
- (b) Deteksi bahasa yang cerdas: Menyoroti dan memeriksa sintaks, mendeteksi beberapa bahasa dalam file.
- (c) Pembagian penampilan dan banyak jendela: Membagi sesi untuk mengedit berdampingan atau dengan beberapa jendela.
- (d) Penyorotan variabel: Klik pada variabel untuk melihat semua kejadian.
- (e) Kotak alat: Potongan simpanan, makro atau perintah shell untuk memudahkan akses.
- (f) Cuplikan singkatan secara otomatis: Ketik singkatan untuk secara otomatis memicu pilihan potongan.
- (g) Pergi ke definisi: Langsung ke definisi variabel, class, dll.
- (h) Kunci pengikatan: Vi / Vim, OS X, emacs, termasuk Windows / Linux.
- (i) Makro dan cuplikan yang powerfull: Masukkan potongan dengan singkatan. Gunakan makro untuk membuat sekuen berulang secara otomatis.
- (j) Tinjauan browser: Pratinjau di jendela terpisah, di editor panel, atau dalam tampilan split.
- (k) Perluas / sesuaikan: Kunci binding, skema warna sintaks, pemeriksa sintaks, format secara otomatis.
- (l) Cuplikan dengan kondisi: Gunakan javascript tertanam untuk menentukan isi potongan saat runtime.

### **Pengkodean Cerdas**

- (a) Calltips dan mengisi secara otomatis: Mendukung banyak bahasa. Mendeteksi beberapa bahasa dalam file.
- (b) Karakter perangkat lunak, pencocokan pasangan: mengisi secara otomatis pasangan kode, dan menyoroti pasangan kode yang cocok. Memberi spasi secara otomatis dan memformat secara otomatis.
- (c) Dukungan perpustakaan kelompok 3d: Termasuk katalog API mengaktifkan mengisi secara otomatis dan calltips untuk perpustakaan seperti PyWin32 dan jQuery.

- (d) Daftar bagian: Melihat dan menyaring daftar fungsi Anda, kelas, atau element.
- (e) Kode browser: Arahkan struktur sumber kode secara hirarki, penampilan dilipat.
- (f) Penampil DOM: Memeriksa dan mengedit pohon DOM untuk dokumen XML dan HTML.
- (g) Dukungan XML: Mengisi secara otomatis untuk dialek XML dapat ditambahkan melalui DTD atau skema RelaxNG.

### **Debug grafis**

- (a) Debug grafis: PHP 4.4 - PHP 5.4, Python 2.4 - Python 3.3, Perl 5.8 - Perl 5.16, Ruby 1.8 - Ruby 1.9, Tcl 8.4 sampai 8.6, Node.js Konfigurasi break-point: Set properti seperti jeda dalam file, garis, kondisi, fungsi panggil dan kembali, pengecualian, dll. Lokal dan remote: Debug program berjalan pada server remote atau platform yang tidak mendukung.
- (b) RX Toolkit: Membuat dan men-debug ekspresi reguler.
- (c) Konsolidasi tampilan: Melihat output, memanggil stack, melihat variabel dalam satu panel.
- (d) Debug multithreaded: Memecah beberapa thread. Engineer dapat menggunakan komodo untuk debug kode python.
- (e) Debug multi proses: Debug beberapa sesi secara bersamaan.
- (f) Pemeriksa HTTP: Pemeriksa komunikasi peramban server, yaitu: ajax / HTTP.
- (g) Penyorotan variabel berubah: Menyoroti secara otomatis variabel yang berubah setelah breakpoint.
- (h) Dibangun pada protokol DBGp: Ditulis oleh ActiveState dan saat ini digunakan dalam banyak IDE.

### **Versi Kontrol dan Alat Terpadu**

- (a) Sistem pengendali versi: Dukungan untuk Subversion, Git, Mercurial, Perforce, CVS, dan Bazaar. Menambah / menghapus file, melihat diffs, melihat sejarah revisi, melakukan perubahan, mengembalikan, dan mendorong menyimpan secara jarak jauh untuk sistem kontrol sumber kode terdistribusi.
- (b) Shell interaktif: tersedia untuk Python, Perl, Ruby dan Tcl. Penerbitan: Menjaga direktori lokal dan remote dalam sinkronisasi.
- (c) Profiling kode: Menganalisis kinerja untuk python (2x dan 3x) dan PHP (5.2 atau keatas) dengan dukungan untuk membaca Callgrind / Cachegrind dan format pstats.
- (d) Unit pengujian terintegrasi: Mendukung kerangka utama seperti PHPUnit, Python unittest, Perl TAP, dan Ruby's rake test.

- (e) Database Explorer: Periksa isi database untuk SQLite. Ekstensi yang tersedia untuk MySQL dan Oracle. Perl Dev Kit, Tcl Dev Kit oleh ActiveState: Menyebarkan Perl dan Tcl dalam aplikasi Komodo. Dapatkan lebih dengan Pro Studio.
- (f) Stackato oleh ActiveState: Debug aplikasi dalam lingkungan layanan platform yang kuat.

### **Workflow dan Manajemen File**

- (a) Proyek dan tempat: Mengelola proyek Anda di sidebar kiri. Lihat sistem file Anda dikategorikan oleh proyek.
- (b) Buka file di jendela samping baru: Jendela samping baru mengatur semua file yang terbuka berdasarkan bahasa program.
- (c) Navigasi minimap: Reposisi manapun dalam kode Anda dari tampilan tingkat tinggi.
- (d) Sinkronisasi komodo: Sinkronisasi item proyek dan preferensi di semua workstation Anda.
- (e) Terbuka secara cepat: Mencari file dengan menekan tombol secara minimal.
- (f) Shortcut terbuka cepat: Buat jalan pintas ke jalur file yang panjang.
- (g) Sejarah, Bookmark, Brace: Melompat ke baris sebelumnya dan ditandai, dan penguat yang cocok.
- (h) Panel notifikasi: Melacak semua proses di satu tempat.

### **Dukungan Bahasa Program**

- (a) Pengeditan: CoffeeScript, CSS, HTML, JavaScript, LESS, Node.js, Perl, PHP, Python, Ruby, SCSS, Tcl, XSLT, Macam-macam template web, dan masih banyak lagi.
- (b) Pengkodean cerdas: CSS, HTML, JavaScript, Node.js, Perl, PHP, Python, Ruby, Tx1, XSLT, Macam-macam template web, dan masih banyak lagi.
- (c) Debug: Node.js, Perl, PHP, Python, Ruby, Tcl, XSLT.

## **5. Atom**



**Gambar 2.32** Logo Atom



Atom adalah text editor yang bersifat free source dan open source bagi macOS, Linux, dan Microsoft Windows. Atom dilengkapi dengan plug-in yang ditulis dalam Node.js yang dikembangkan oleh GitHub . Atom merupakan aplikasi desktop yang dibuat memakai teknologi web.

Fitur-fitur yang tersedia di dalam Atom antara lain teknologi web based, fuzzy finder, fast project-wide, file system browser, TextMate grammars, integrasi dengan Node.js, dan masih banyak keunggulan lain Atom dibandingkan dengan text editor lain.

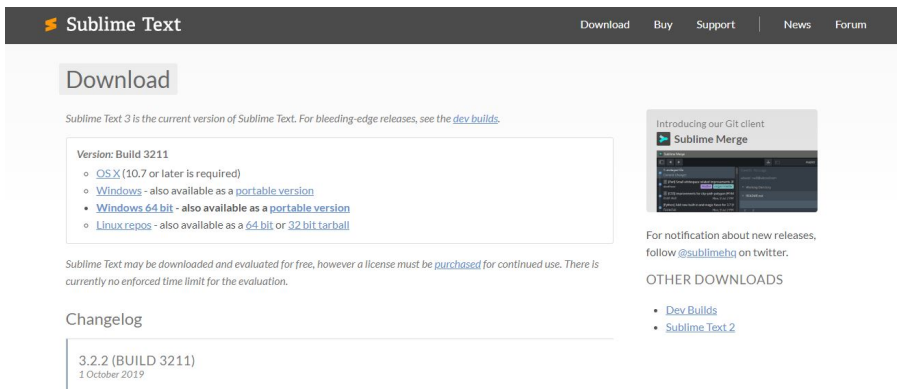
### **Keunggulan Atom**

- (a) Keunggulan text editor satu ini yang pertama adalah Atom merupakan salah satu aplikasi open source alias gratis. Dengan menggunakan Atom Anda tidak akan dituntut untuk membeli lisence product ini dengan tarif tertentu.
- (b) Keunggulan dari Text Editor Atom yang kedua adalah, dengan menggunakan text editor Atom Anda sebagai programmer bisa langsung terkoneksi dengan project Anda yang ada di Github. Tentunya jika Anda menginginkannya, Anda sudah harus melakukan instalasi git svn.
- (c) Yang ketiga, dengan menggunakan text editor Atom, Anda bisa dengan mudah melakukan instalasi packages yang Anda butuhkan demi mendukung performa dan kebutuhan Anda sebgai programmer dalam menulis code.
- (d) Text editor Atom juga menyediakan banyak theme yang dapat Anda gunakan untuk mempercantik text editor kesayangan agar Anda tidak bosan saat menulis dan melihat lautan teks coding.
- (e) Ada juga fitur customize styling yang artinya Anda bebas merubah settingan Atom yang mau Anda pakai, karena pada dasarnya memang atom dibuat menggunakan electron dan javascript, jadi untuk style sama seperti Anda menulis file css.

### **2.5.3 Install Submlie Text 3**

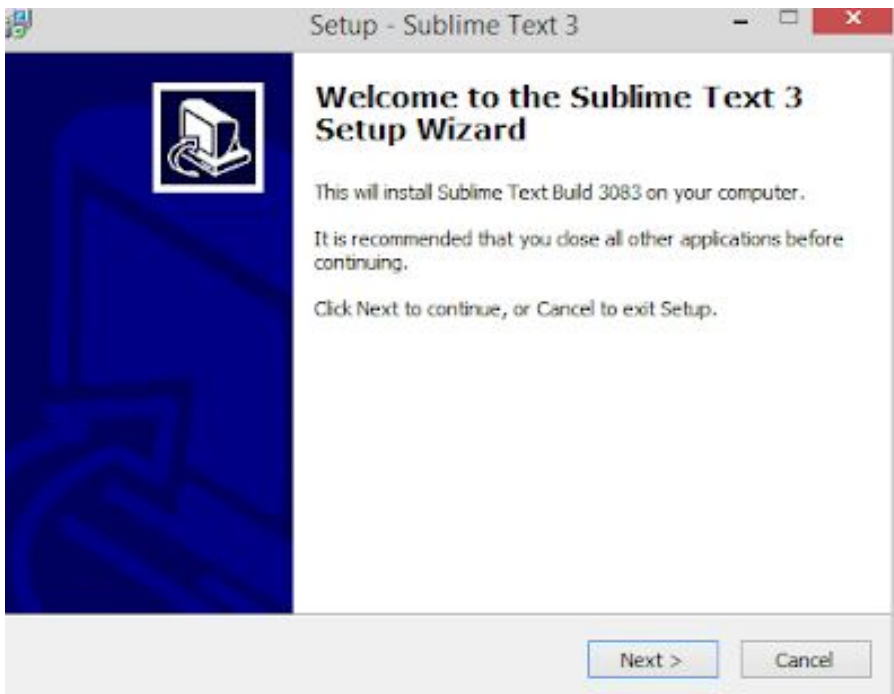
langkah-langkah untuk install Sublime 3 sebagai berikut:

1. masuk ke web Sublime 3 <https://www.sublimetext.com/3> lalu klik Download



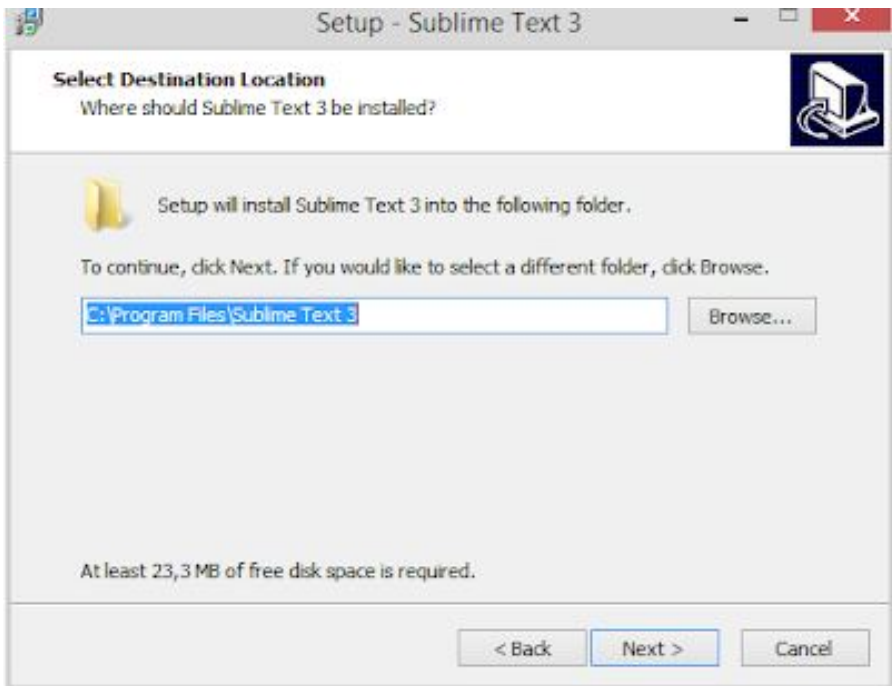
**Gambar 2.33** Langkah pertama

2. setelah selesai download , klik 2x pada file , sehingga muncul seperti ini



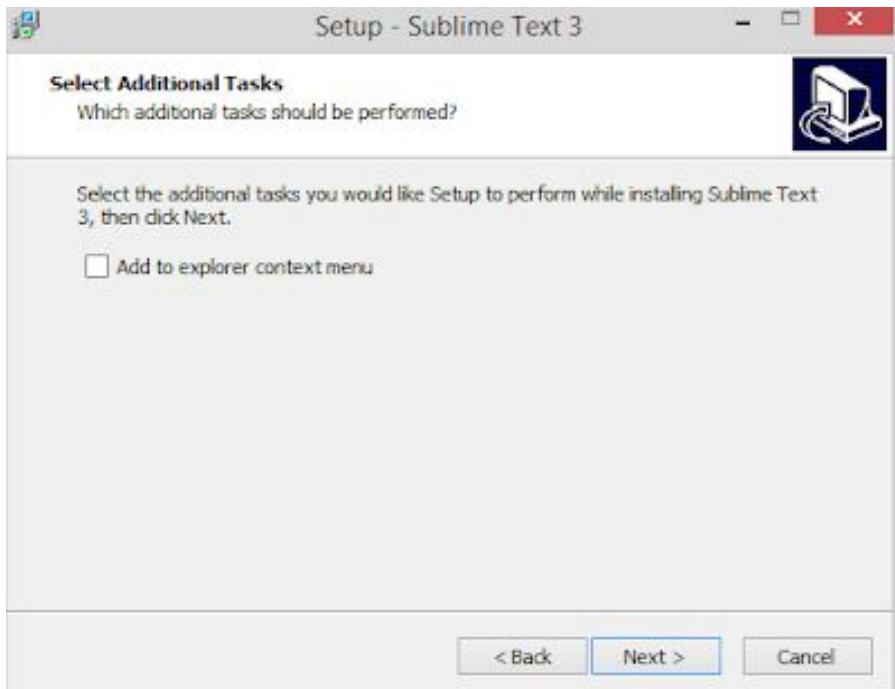
**Gambar 2.34** Langkah Kedua

3. setelah itu klik tombol next



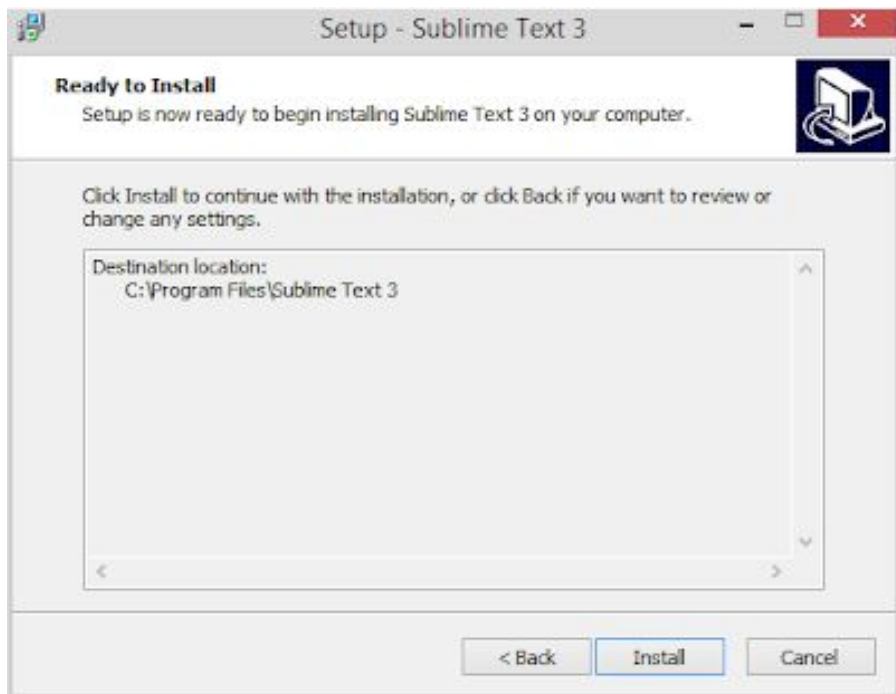
**Gambar 2.35** Langkah Ketiga

4. lalu tempatnya file sublime tergantung anda ingin simpan dimana , biasanya di simpan di file C.



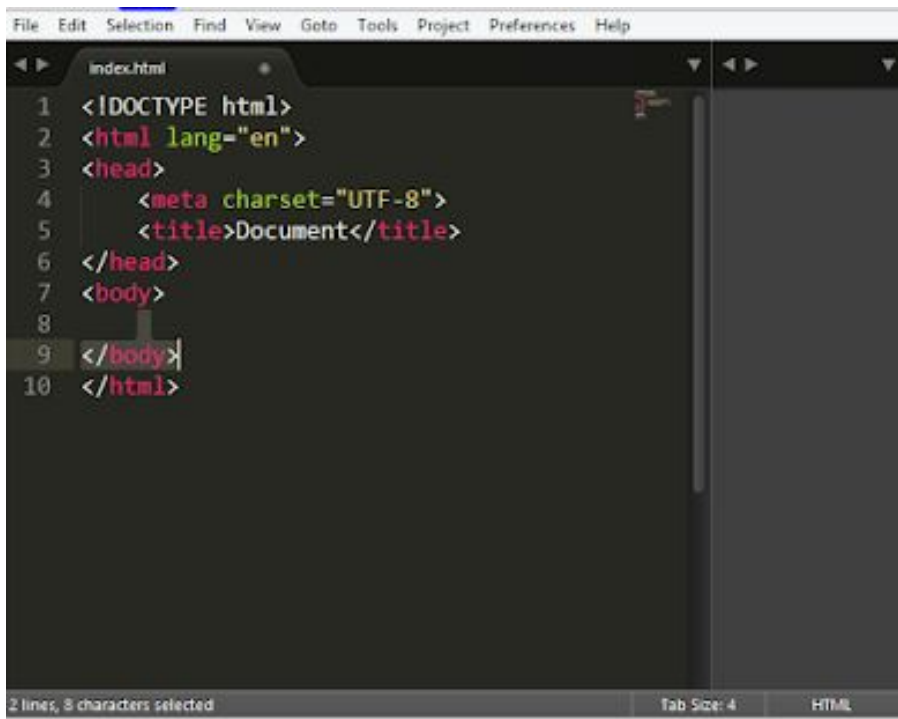
**Gambar 2.36** Langkah Keempat

5. lalu klik next



**Gambar 2.37** Langkah Kelima

6. maka apabila selesai maka akan tampil seperti di bawah ini dan siap untuk di gunakan



**Gambar 2.38** Halaman Sublime Text



## BAB 3

---

# PERANCANGAN SISTEM

---

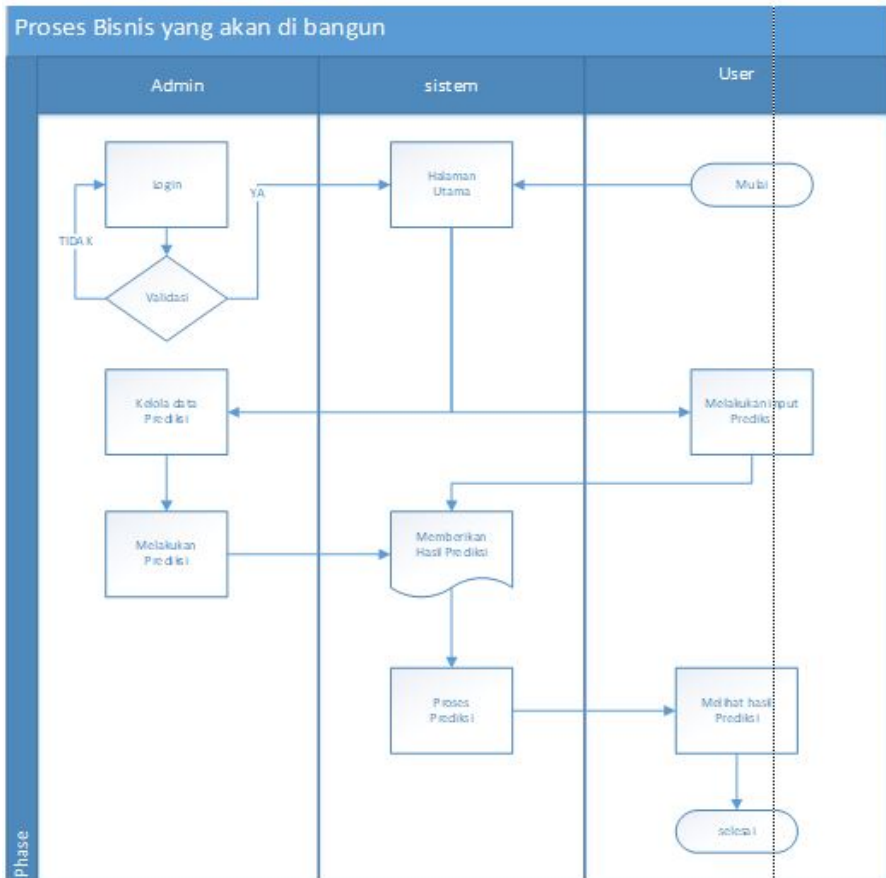
### 3.1 Analisis

#### 3.1.1 Pengertian Analisi

Pada analisi ini penulis menganalisis tentang aplikasi yang akan di bangun, sehingga dengan menganalisis dapat mengidentifikasi dan mengevaluasi permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga diusulkan perbaikan-perbaikan serta pembentukan dan penyusunan suatu sistem untuk dikaji lebih lanjut. Sebelum merancang sebuah aplikasi yang akan dibangun, faktor yang harus diperhatikan yaitu adalah tahap analisis. Menggunakan suatu strategi yang merupakan metodologi yang dirasa cukup efektif dalam merancang aplikasi. Metodologi yang digunakan adalah Metode Moving Average digunakan untuk melakukan peramalan, Rata-rata bergerak (Moving Average) adalah suatu metode peramalan yang dilakukan dengan mengambil sekelompok nilai pengamatan, mencari nilai rata-rata tersebut sebagai ramalan untuk periode yang akan datang (Subagyo, 2008).



### 3.1.2 Analisis Proses Bisnis yang akan dibangun



**Gambar 3.1** FlowChart Analisi yang akan DiBangun

Keterangan:

1. user memasukkan halaman utama. Lalu ke halaman Prediksi
2. Di halaman Prediksi, user/Admin dapat memasukkan jumlah hari yang ingin di prediksi
3. Prediksi akan diproses disistem.
4. Hasilnya akan muncul di halaman prediksi
5. Hasilnya dapat di cetak dalam bentuk PDF.

6. Admin terlebih dahulu melakukan login, system akan mengecek atau memvalidasi apakah username dan password yang dimasukkan oleh admin benar, jika iya maka system akan menampilkan halaman utama. Jika tidak, kembali ke halaman login.
7. Setelah login, admin masuk ke halaman utama web. Terdapat menu-menu yang dapat dijalankan oleh admin seperti kelola data.

### 3.1.3 Kebutuhan Aplikasi

Kebutuhan aplikasi merupakan suatu cara agar dapat menghasilkan aplikasi yang sesuai dengan aplikasi yang dibutuhkan. Aplikasi yang dibuat terdiri dari:

1. Perangkat Lunak (Software)
2. Perangkat Keras (Hardware)

**3.1.3.1 Perancangan Arsitektur Perangkat Lunak dan Perangkat Keras** Adapun spesifikasi perangkat keras (Hardware) yang digunakan saat pembuatan aplikasi adalah sebagai berikut:

No	Nama Perangkat	Spesifikasi	Keterangan
1	<i>Memory</i>	4 GB	<i>Memory System</i> yang digunakan
2	<i>Processor</i>	<i>Intel® core™ i7-5500U CPU @ 2,39 GHz</i>	Untuk kecepatan transfer data dari sistem yang sangat bergantung pada kecepatan prosesor komputer

**Gambar 3.2** Perangkat Keras

**3.1.3.2 Perangkat Lunak** Adapun spesifikasi perangkat lunak (Software) yang digunakan untuk membuat aplikasi adalah sebagai berikut:

No	Tools / Software	Fungsi	Keterangan
1.	<i>Windows 10</i>	<i>Sistem Operasi</i>	-
2.	<i>Star UML</i>	<i>Pembuatan Diagram UML</i>	-
3.	<i>Google Chrome</i>	<i>Browser</i>	-

**Gambar 3.3** Perangkat Lunak

## 3.2 Design Aplikasi

Pada tahapan ini terdapat beberapa proses yaitu use case diagram, class diagram, dan activity diagram.

### 3.2.1 *Use Case diagram*

Use case diagram ini menunjukkan apa saja yang dapat dilakukan oleh aktor pada aplikasi Prediksi. Pada use case tersebut masing-masing aktor memiliki cara pengaksesan yang berbeda. Usecase Diagram menggambarkan alur sistem secara ringkas dan menggambarkan kebutuhan fungsionalitas yang diharapkan oleh sebuah sistem.

Use Case merupakan sebuah teknik yang digunakan dalam pengembangan sebuah software atau sistem informasi untuk menangkap kebutuhan fungsional dari sistem yang bersangkutan, Use Case menjelaskan interaksi yang terjadi antara aktor inisiator dari interaksi sistem itu sendiri dengan sistem yang ada, sebuah Use Case direpresentasikan dengan urutan langkah yang sederhana.

Perilaku sistem adalah bagaimana sistem beraksi dan bereaksi. Perilaku ini merupakan aktifitas sistem yang bisa dilihat dari luar dan bisa diuji. Perilaku sistem ini di-capture di dalam USE CASE. USE CASE sendiri mendeskripsikan sistem, lingkungan sistem, serta hubungan antara sistem dengan lingkungannya.

Deskripsi dari sekumpulan aksi sekuensial yang ditampilkan sistem yang menghasilkan yang tampak dari nilai ke actor khusus. Use Case digunakan untuk menyusun behavioral things dalam sebuah model. Use case direalisasikan dengan sebuah collaboration. Secara gambar, sebuah use case digambarkan dengan sebuah ellips dengan garis penuh.

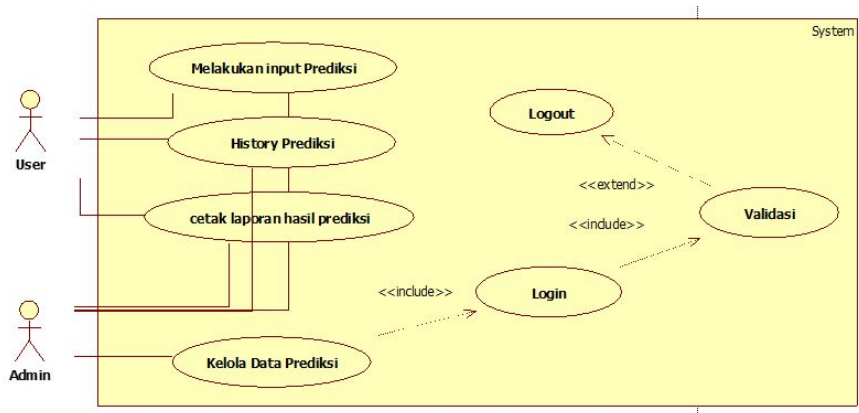
#### **Manfaat Use Case**

1. Digunakan untuk berkomunikasi dengan end user dan domain expert.
2. Memastikan pemahaman yang tepat tentang requirement / kebutuhan sistem.
3. Digunakan untuk mengidentifikasi siapa yang berinteraksi dengan sistem dan apa yang harus dilakukan sistem.
4. Interface yang harus dimiliki sistem.
5. Digunakan untuk verifikasi.

#### **Karakteristik**

1. Use cases adalah interaksi atau dialog antara sistem dan actor, termasuk pertukaran pesan dan tindakan yang dilakukan oleh sistem.
2. Use cases diprakarsai oleh actor dan mungkin melibatkan peran actor lain. Use cases harus menyediakan nilai minimal kepada satu actor.
3. Use cases bisa memiliki perluasan yang mendefinisikan tindakan khusus dalam interaksi atau use case lain mungkin disisipkan.
4. Use case class memiliki objek use case yang disebut skenario. Skenario menyatakan urutan pesan dan tindakan tunggal.

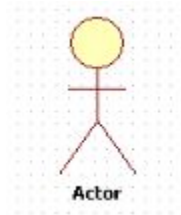
Berikut Usecase Diagram System



**Gambar 3.4** Usecase

Use Case memiliki beberapa komponen yaitu :

### 1. Aktor



**Gambar 3.5** Aktor

Menggambarkan seseorang yang berinteraksi dengan sistem, di mana hanya bisa menginputkan informasi dan menerima informasi dari sistem dan tidak memegang kendali pada use case. Dan biasa actor di gambarkan dengan stickman.

### 2. Use Case



**Gambar 3.6** Komponen Usecase

Gambaran fungsional sistem yang akan di buat, agar pengguna lebih mengerti penggunaan system.

### 3. Relasi Dalam *Use Case*

Ada beberapa relasi yang terdapat pada use case diagram:

- (a) *Association* menghubungkan link antar element.
- (b) *Generalization* sebuah elemen yang menjadi spesialisasi dari elemen yang lain.
- (c) *Dependency* sebuah elemen yang bergantung beberapa cara kepada elemen lainnya.
- (d) *Aggregation* bentuk assosiation yang di mana sebuah elemen berisi elemen lainnya.

Tipe relasi/ *stereotype* yang mungkin terjadi pada *use case diagram*:

- (a) `include`, yaitu kelakuan yang harus terpenuhi agar sebuah event dapat terjadi, dimana pada kondisi ini sebuah use case adalah bagian dari use case lainnya.
- (b) `extends`, kelakuan yang hanya berjalan di bawah kondisi tertentu seperti menggerakkan alarm.
- (c) `communicates`, mungkin ditambahkan untuk asosiasi yang menunjukkan asosiasinya adalah communicates association . Ini merupakan pilihan selama asosiasi hanya tipe relationship yang dibolehkan antara actor dan use case.

### 3.2.2 Definisi *Use Case* dan Aktor

No	Aktor	Deskripsi
1.	<i>User</i>	<i>User</i> adalah orang yang memiliki hak akses untuk melakukan prediksi.
2.	<i>Admin</i>	<i>Admin</i> adalah orang yang memiliki hak akses untuk melakukan kelola data yang ada pada sistem.

**Gambar 3.7** Definisi Aktor

No	Use Case	Deskripsi
1.	Login	Merupakan proses untuk melakukan identifikasi pengguna sistem dengan memasukkan <i>username</i> dan <i>password</i> .
2.	Melakukan input Prediksi	Merupakan proses yang dilakukan oleh <i>user</i> untuk melakukan <u>Prediksi</u> /memasukkan prediksi ke form <u>prediksi</u> .
3	Cetak laporan hasil prediksi	Merupakan proses yang di lakukan oleh <i>User dan admin</i> berupa data dalam bentuk laporan yang bersisi data hasil <u>prediksi</u> .
4.	History prediksi	Merupakan proses yang di lakukan oleh <i>user dan admin</i> untuk melihat <u>history prediksi yang sudah di lakukan</u> .
5.	Kelola data	Merupakan proses yang dilakukan oleh <i>admin</i> untuk mengelola data .
6.	Logout	Merupakan proses untuk keluar dari sistem sebagai pengguna sistem.

Gambar 3.8 Definisi Use Case

### 3.2.3 Skenario Use Case

skenario *use case* / *use case* skenario adalah alur jalannya proses use case dari sisi aktor dan system.

Skenario use case dibuat per use case terkecil, misalkan untuk generalisasi maka scenario yang dibuat adalah use case yang lebih khusus. Skenario normal adalah scenario bila system berjalan normal tanpa terjadi kesalahan atau error. Sedangkan skenario alternatif adalah scenario bila system tidak berjalan normal atau mengalami error. Skenario normal dan skenario alternatif dapat berjumlah lebih dari satu. Alur skenario inilah yang nantinya menjadi landasan pembuatan sequence diagram / diagram sekuen.

Berikut adalah format tabel skenario use case.

#### 1. Skenario Use Case Login

Identifikasi	
Nomor	UC1
Nama	<i>Login</i>
Tujuan	Untuk masuk ke dalam sistem
Deskripsi	
Aktor	<i>Admin</i>
Skenario Utama	
Kondisi Awal	Masuk <i>Form Login</i>
Aksi Aktor	Reaksi Sistem
1. <i>User</i> memasukkan <i>username</i> dan <i>password</i>	2. Sistem akan menerima masukkan <i>username</i> dan <i>password</i>
3. <i>User</i> harus menekan tombol <i>login</i>	4. Sistem akan memvalidasi <i>username</i> dan <i>password</i> . Apabila <i>username</i> dan <i>password</i> benar maka akan muncul halaman utama. Jika salah, maka sistem akan meminta ulang <i>username</i> dan <i>password</i>
Kondisi Akhir	Jika <i>username</i> dan <i>password</i> yang dimasukkan oleh <i>user</i> tidak sesuai maka akan muncul <i>form login</i> kembali.

**Gambar 3.9** Skenario *Use Case* login

keterangan:

- (a) Use case :Login
- (b) Actor :*Admin*
- (c) Pre-condition :actor ingin menggunakan program and belum melakukan login
- (d) Post-condition :actor telah melakukan login dan berinteraksi dengan program
- (e) Description :actor melakukan login untuk dapat menggunakan program

## 2. Skenario *Use Case* Melakukan input Prediksi

Identifikasi	
Nomor	UC2
Nama	Melakukan <u>prediksi</u>
Tujuan	Memasukkan <u>angka prediksi</u> kedalam sistem.
Deskripsi	
Aktor	User
Skenario Utama	
Kondisi Awal	Masuk ke Halaman <u>prediksi</u>
Aksi Aktor	Reaksi Sistem
1. User memilih menu <u>cetak laporan</u>	2. System akan menampilkan data <u>laporan</u> hasil <u>prediksi</u> .
3. User <u>megklik cetak</u>	4. system akan menampilkan <u>halaman dalam bentuk PDF</u> dan dapat di download
Kondisi Akhir	Sistem akan menampilkan <u>halaman baru dalam bentuk PDF</u> .

**Gambar 3.10** Skenario Use Case Melakukan input Prediksi

keterangan:

- (a) Use case :Melakukan Input Prediksi
- (b) Actor :Admin/User
- (c) Pre-condition :actor ingin melakukan prediksi
- (d) Post-condition :actor telah melakukan prediksi dan berhasil melihat hasil prediksi
- (e) Description :actor melakukan prediksi agar melihat hasil dari prediksi tersebut

### 3. Skenario Use Case Cetak laporan hasil Prediksi

Identifikasi	
Nomor	UC3
Nama	<u>Cetak laporan hasil Prediksi</u>
Tujuan	<u>Mencetak laporan hasil Prediksi</u>
Deskripsi	
Aktor	User/Admin
Skenario Utama	
Kondisi Awal	Masuk ke Halaman <u>prediksi</u>
Aksi Aktor	Reaksi Sistem
1. User dan Admin memilih menu <u>prediksi</u> .	2. User melakukan input <u>prediksi</u> .
3. Sistem akan menampilkan hasil prediksi.	
Kondisi Akhir	Sistem akan menampilkan hasil prediksi.

**Gambar 3.11** Skenario Use Case Cetak laporan hasil Prediksi



keterangan:

- (a) Use case :Cetak Laporan Hasil Prediksi
- (b) Actor :*Admin/User*
- (c) Pre-condition :actor ingin mencetak hasil prediksi
- (d) Post-condition :actor telah mencetak prediksi dan dapat di Import dalam bentuk PDF.
- (e) Description :actor melakukan cetak agar dapat melihat hasil dari prediksi

#### 4. Skenario *Use Case History* Prediksi

Identifikasi	
Nomor	UC3
Nama	History Prediksi
Tujuan	Melihat prediksi yang sudah dilakukan
Deskripsi	
Aktor	User/Admin
Skenario Utama	
Kondisi Awal	Masuk ke Halaman history prediksi
Aksi Aktor	Reaksi Sistem
1. User dan admin memilih menu history prediksi.	2. User dan admin melihat history prediksi
Kondisi Akhir	Sistem akan menampilkan history prediksi

**Gambar 3.12** Skenario *Use Case History* Prediksi

keterangan:

- (a) Use case :*History* Prediksi
- (b) Actor :*Admin/User*
- (c) Pre-condition :actor ingin melihat *history* hasil prediksi
- (d) Post-condition :actor telah melakukan prediksi dan dapat melihat hasilnya kembali di *history* prediksi.
- (e) Description :actor dapat melihat *history* prediksi setelah melakukan prediksi

#### 5. Skenario *Use Case* Kelola Data

Identifikasi	
Nomor	UC4
Nama	Kelola Data
Tujuan	Mengelola data. Kelola data yaitu menambah, mengubah, dan menghapus data pengaduan.
Deskripsi	
Aktor	<i>Admin</i>
Skenario Utama	
Kondisi Awal	Masuk ke halaman Kelola Data.
Aksi Aktor	Reaksi Sistem
1. <i>Admin</i> memilih menu Kelola Data	2. Sistem akan menampilkan menu kelola data.
3. <i>Admin</i> memasukkan data.	4. Sistem akan melakukan verifikasi data dan menyimpan data di dalam <i>database</i> .
5. <i>Admin</i> mengubah data pengaduan.	6. Sistem akan melakukan verifikasi data dan mengubah data di dalam <i>database</i> .
7. <i>Admin</i> menghapus data.	8. Sistem akan melakukan verifikasi data dan menghapus data di dalam <i>database</i> .
Kondisi Akhir	Kelola Data Berhasil

**Gambar 3.13** Skenario *Use Case* Kelola Data

keterangan:

- (a) Use case :Kelola Data
- (b) Actor :*Admin*
- (c) Pre-condition :aktor ingin melakukan kelola data
- (d) Post-condition :aktor telah melakukan kelola data yang dapat di lihat detail, tambah, diperbarui, dan di hapus.
- (e) Description :aktor dapat melakukan kelola data setelah login dan mendapatkan hak akses

#### 6. Skenario *Use Case Logout*.

Identifikasi	
Nomor	UC5
Nama	<i>Logout</i>
Tujuan	Mengeluarkan akun pengguna sistem ketika pengguna telah selesai menggunakan sistem.
Deskripsi	
Aktor	<i>admin.</i>
Skenario Utama	
Kondisi Awal	Masuk <i>form login</i> .
Aksi Aktor	Reaksi Sistem
1. <i>admin</i> menekan tombol <i>logout</i> .	2. Sistem akan mengeluarkan akun pengguna.
Kondisi Akhir	<i>User/admin</i> akan kembali ke halaman utama atau <i>halaman login</i> .

**Gambar 3.14** Skenario *Use Case Logout*.

keterangan:








- (a) Use case :*logout*
- (b) Actor :*Admin*
- (c) Pre-condition :actor ingin keluar dari hak akses yang dimiliki
- (d) Post-condition :actor telah melakukan keluar dari hak akses.
- (e) Description :actor setelah keluar tidak dapat melakukan kelola data setelah keluar dan kembali ke halaman utama.

### 3.2.4 *Class diagram*

*Class diagram* adalah diagram yang menggambarkan struktur sistem dari segi pen-definisian kelas-kelas yang akan dibuat untuk membangun sistem. kelas memiliki 3 bagian utama yaitu attribute, operation, dan name. kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem. Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis kelas berikut :

1. Kelas Main. Yaitu kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.
2. Kelas Interface. Kelas yang mendefinisikan dan mengatur tampilan ke pemakai. Biasanya juga disebut kelas boundaries.
3. Kelas yang diambil dari pendefinisian usecase. Merupakan kelas yang menangani fungsi-fungsi yang harus ada dan diambil dari pendefinisian usecase.
4. Kelas Entitas. Merupakan kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data.

Relasi yang digunakan dalam penggambaran kelas diagram dapat dilihat dalam tabel berikut :

Simbol	Keterangan
<b>Asosiasi/Assosiation</b> 	Hubungan statis antar kelas. Asosiasi menggambarkan kelas yang memiliki atribut berupa kelas lain, atau kelas yang harus mengetahui eksistensi kelas lain. Asosiasi biasanya disertai dengan <i>multiplicity</i>
<b>Asosiasi berarah/Directed Assosiation</b> 	Asosiasi dengan makna kelas yang satu digunakan oleh kelas yang lain. Asosiasi berarah juga biasanya disertai dengan <i>multiplicity</i>
<b>Generalization</b> 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus) atau untuk menyatakan hubungan <i>inheritance</i> .
<b>Dependency/kebergantungan</b> 	Relasi antar kelas dengan makna kebergantungan antar kelas.
<b>Agregasi/Agregation</b> 	Hubungan yang menyatakan bahwa suatu kelas menjadi atribut bagi kelas lain
<b>Komposisi/Composition</b> 	Bentuk khusus dari agregasi dimana kelas yang menjadi bagian diciptakan setelah kelas menjadi <i>whole</i> dibuat. Misal kelas <i>whole</i> dihapus, maka kelas yang menjadi <i>part</i> ikut musnah
<b>Realization</b> 	Hubungan antar kelas dimana sebuah kelas memiliki keharusan untuk mengikuti aturan yang ditetapkan oleh kelas lainnya.

**Gambar 3.15** Relasi *Class diagram*

Dalam suatu diagram kelas atribut dan method dapat memiliki salah satu sifat berikut : Private : tidak dapat dipanggil dari luar kelas yang bersangkutan Protected : Hanya dapat dipanggil oleh kelas yang bersangkutan dan anak-anak kelas yang mewarisinya. Public : Dapat dipanggil oleh siapa saja.

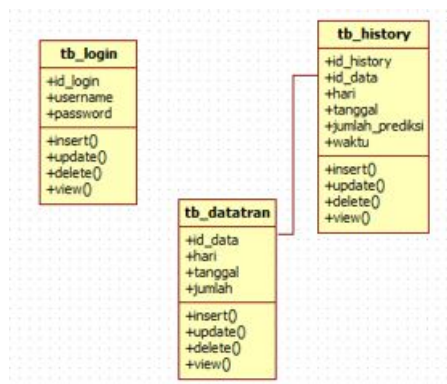
1. Private : tidak dapat dipanggil dari luar kelas yang bersangkutan
2. Protected : Hanya dapat dipanggil oleh kelas yang bersangkutan dan anak-anak kelas yang mewarisinya.
3. Public : Dapat dipanggil oleh siapa saja.

Multiplicity adalah jumlah banyaknya objek sebuah kelas yang berelasi dengan sebuah objek lain dari kelas lain yang berasosiasi dengan kelas tersebut. Macam-macam multiplicity disajikan dalam tabel berikut :

Multiplicity	Arti
*	Banyak
0	Tepat 0
1	Tepat satu
0...*	Nol atau lebih
1...*	Satu atau lebih

**Gambar 3.16** Macam-Macam Multiplicity

Berikut ini *Class Diagram* Aplikasi Prediksi .



**Gambar 3.17** *Class diagram* Aplikasi Prediksi

### 3.2.5 Sequence Diagram

Sequence Diagram adalah salah satu dari diagram - diagram yang ada pada UML, sequence diagram adalah diagram yang menggambarkan kolaborasi dinamis antara sejumlah object. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara object juga interaksi antara object. Sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem.

Dalam UML, object pada sequence diagram digambarkan dengan segi empat yang berisi nama dari object yang digarisbawahi. Pada object terdapat 3 cara untuk menamainya yaitu : nama object, nama object dan class, dan nama class. Berikut contoh dari ketiga cara tersebut :

Dalam sequence diagram, setiap object hanya memiliki garis yang digambarkan garis putus-putus ke bawah. Pesan antar object digambarkan dengan anak panah dari object yang mengirimkan pesan ke object yang menerima pesan. **Komponen-Komponen Sequence Diagram**

1. Object - adalah komponen berbentuk kotak yang mewakili sebuah class atau object. Mereka mendemonstrasikan bagaimana sebuah object berperilaku pada sebuah system.
2. Activation boxes - adalah komponen yang berbentuk persegi panjang yang menggambarkan waktu yang diperlukan sebuah object untuk menyelesaikan tugas. Lebih lama waktu yang diperlukan, maka activation boxes akan lebih panjang.
3. Actors - adalah komponen yang berbentuk stick figure. Komponen yang mewakili seorang pengguna yang berinteraksi dengan system.
4. Lifeline - adalah komponen yang berbentuk garis putus - putus. Lifeline biasanya memuat kotak yang berisi nama dari sebuah object. Berfungsi menggambarkan aktifitas dari object.
5. General - adalah Merepresentasikan entitas tunggal dalam sequence diagram . Entitas ini memiliki nama , stereotype atau berupa instance (class).
6. Boundary - Boundary biasanya berupa tepi dari sistem , seperti user interface atau suatu alat yang berinteraksi dengan sistem yang lain.
7. Control - Control elemenr mengatur aliran dari informasi untuk sebuah skenario . Objek ini umumnya mengatur perilaku dan perilaku bisnis.
8. Entity - Entitas biasanya elemen yang bertanggung jawab menyimpan data atau informasi . Ini dapat berupa beans atau model object.
9. Message - sebagai komunikasi antar objek yang menggambarkan aksi yang akan dilakukan . Message terjadi antara dua buah objek dimana satu objek (client) dan meminta objek (supplier) untuk melakukan sesuatu.

### **Penggambaran Diagram Sequence**

Objek - Objek dalam sequence diagram diurutkan dari kiri ke kanan , aktor yang menginisiasi interaksi biasanya ditaruh di paling kiri dari diagram . Sequence diagram terdiri atas 2 dimensi , yaitu :

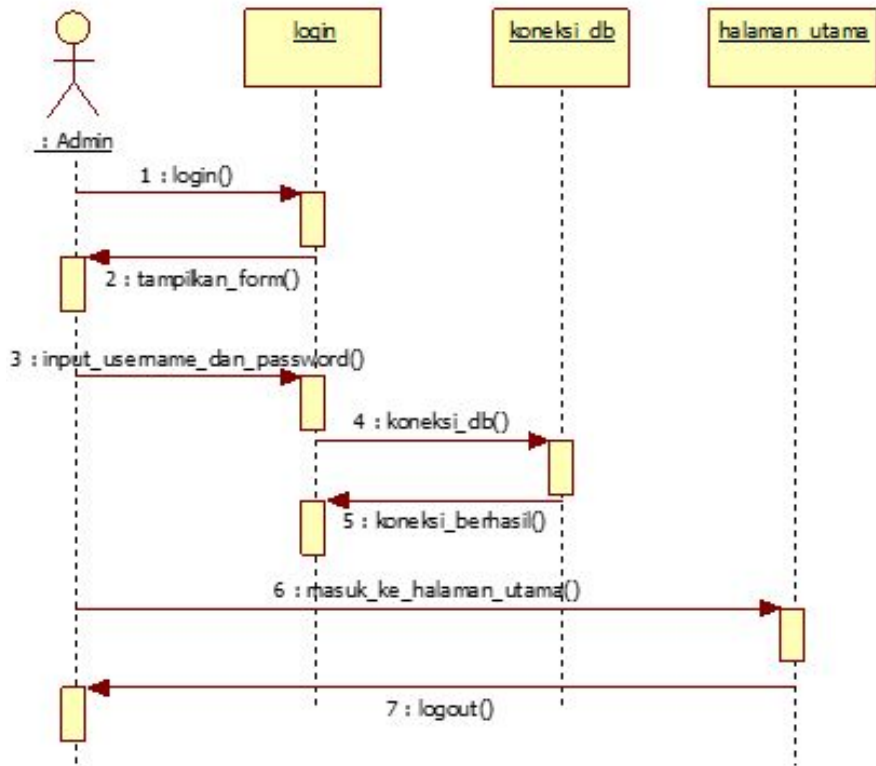
1. Dimensi vertikal (merepresentasikan waktu)
2. Dimensi horizontal (merepresentasikan objek - objek yang terkait).

Bagian paling atas dari diagram menjadi titik awal dan waktu berjalan ke bawah sampai dengan bagian dasar dari diagram . Garis vertical , disebut lifeline , diletakkan pada setiap objek atau aktor . kemudian lifeline tersebut digambarkan menjadi kotak ketika objek melakukan suatu operasi , kotak tersebut disebut dengan activation box . Objek dikatakan mempunyai live activation pada saat tersebut . Pesan

yang dipertukarkan antar objek digambarkan sebagai sebuah anak panah antara activation box pengirim dan penerima . Kemudian di atasnya diberikan label pesan.

Sequence Diagram pada aplikasi prediksi ini seperti sebagai berikut:

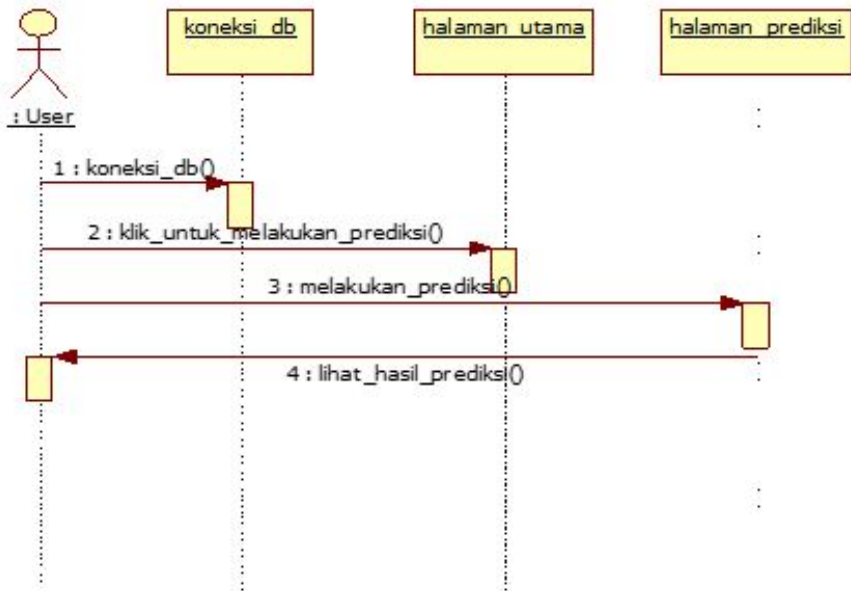
### 1. Sequence Diagram Login



**Gambar 3.18** Sequence Diagram Login

Deskripsi *Sequence Diagram* Proses login dimulai ketika admin memasukkan username dan password di form login akan melakukan passing terhadap username dan password yang telah diinputkan kedalam koneksi database. Database kemudian akan melakukan validasi terhadap username dan password sesuai user yang terintegrasi didalam database. Setelah melakukan proses maka admin akan diarahkan ke halaman utama. Ketika admin melakukan logout, database akan memutuskan koneksi dari halaman utama.

### 2. Sequence Diagram Melakukan Input Prediksi

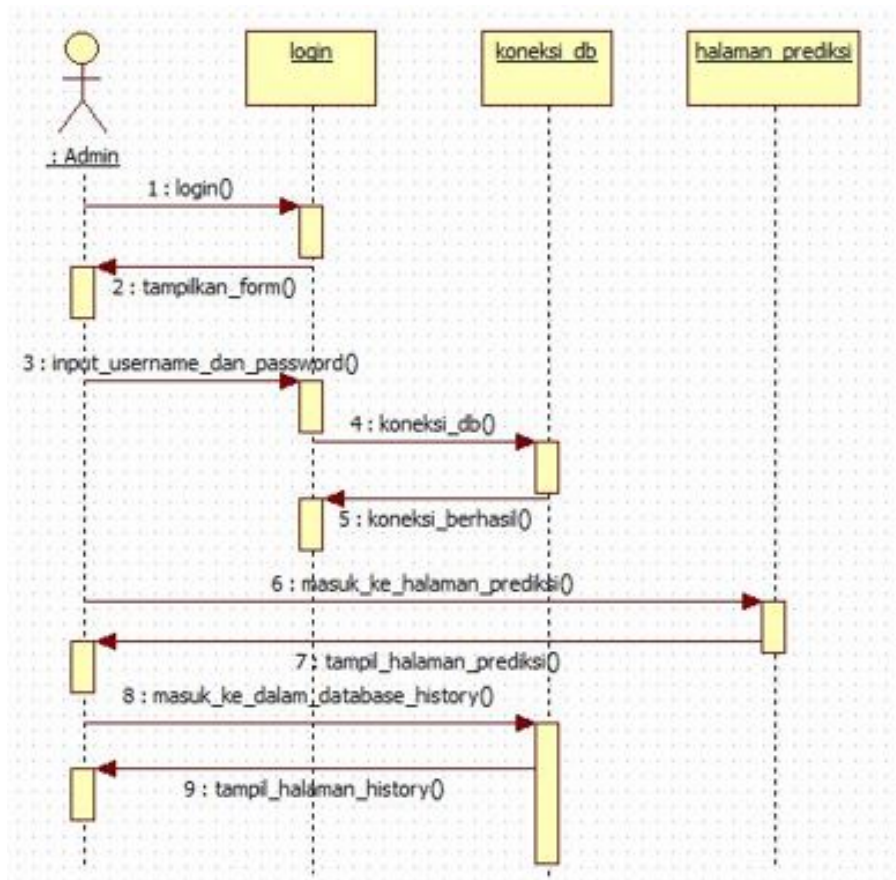


**Gambar 3.19** *Sequence Diagram* Melakukan Input Prediksi

Deskripsi *Sequence Diagram* Proses Melakukan Input Prediksi Pada saat masuk ke halaman utama, akan muncul klik untuk melakukan prediksi, setelah di klik akan masuk ke halaman prediksi di mana akan memasukkan jumlah hari yang akan di prediksi, setelah itu akan muncul jumlah prediksi yang keluar dari system.

### 3. *Sequence Diagram* Cetak laporan hasil Prediksi

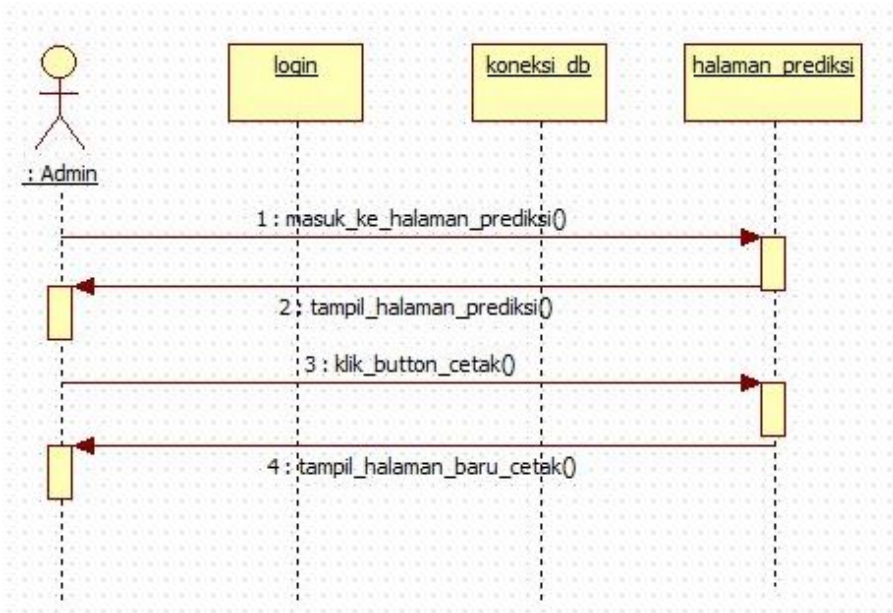




**Gambar 3.20** *Sequence Diagram* Cetak laporan hasil Prediksi

Deskripsi *Sequence Diagram* Proses Cetak laporan hasil Prediksi Proses cetak laporan hasil prediksi dimulai ketika user masuk ke halaman prediksi, setelah melakukan input prediksi akan tampil hasil dan dapat di import dalam bentuk PDF,

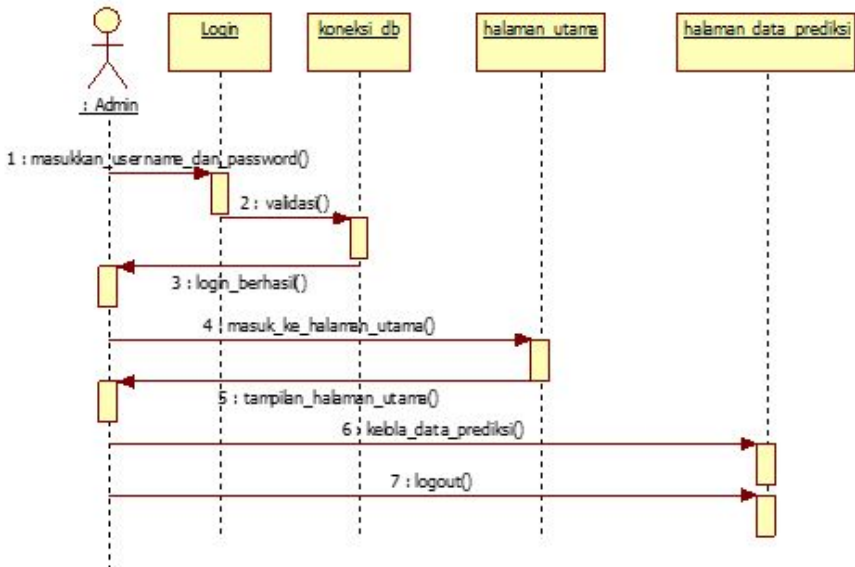
#### 4. *Sequence Diagram History* Prediksi



**Gambar 3.21** *Sequence Diagram History Prediksi*

Deskripsi *Sequence Diagram History Prediksi* Pada saat masuk ke halaman utama , lalu mengklik tombol prediksi, lalu akan masuk ke halaman prediksi, setelah melakukan prediksi , data prediksi akan masuk ke dalam history prediksi yang sewaktu waktu bisa di lihat kembali.

## 5. *Sequence Diagram* Kelola Data



**Gambar 3.22** *Sequence Diagram* Kelola Data

Deskripsi *Sequence Diagram* Proses Kelola Data Proses kelola data dimulai ketika admin memasukkan username dan password di form login, kemudian form login akan melakukan passing terhadap username dan password yang telah diinputkan kedalam koneksi database. Database kemudian akan melakukan validasi terhadap username dan password sesuai admin yang terintegrasi didalam database. Setelah melakukan proses validasi maka akan muncul konfirmasi tergantung validasinya benar atau tidak, jika valid atau benar maka admin akan diarahkan ke halaman utama. Setelah admin di halaman utama, admin dapat memilih halaman data prediksi untuk melakukan pengolahan data.

### 3.2.6 Activity Diagram

Activity Diagram adalah diagram yang menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Komponen yang ada pada activity diagram antara lain :

1. Activity atau state : Menunjukkan aktivitas yang dilakukan.
2. Initial activity atau initial state : Menunjukkan awal aktivitas dimulai.
3. Final Activity atau final state : Menunjukkan bagian akhir dari aktivitas.

4. Decision : Digunakan untuk menggambarkan test kondisi untuk memastikan bahwa control flow atau object flow mengalir lebih ke satu jalur. Jumlah jalur sesuai yang diinginkan.
5. Merge : Berfungsi menggabungkan flow yang dipecah oleh decision.
6. Synchronization : Diabgi menjadi 2 yaitu fork dan join. Fork digunakan untuk memecah behaviour menjadi activity atau action yang paralel, sedangkan join untuk menggabungkan kembali activity atau action yang paralel.
7. Swimlanes : Memecah activity diagram menjadi baris dan kolom untuk membagi tanggung jawab obyek-obyek yang melakukan aktivitas.
8. Transition : Menunjukkan aktivitas selanjutnya setelah aktivitas sebelumnya.

### Fungsi

Activity Diagram adalah Menggambarkan proses bisnis dan urutan aktivitas dalam sebuah proses. Menunjukkan urutan aktifitas proses pada sistem. Activity diagram dibuat berdasarkan sebuah atau beberapa use case pada use case diagram.

### Komponen

Komponen yang ada pada Activity Diagram antara lain :

#### 1. *Start Point/Initial State*(Titik Mulai)

*Start Point* merupakan lingkaran hitam kecil, yang menandakan tindakan awal atau titik awal aktivitas untuk setiap diagram aktivitas.



**Start Point/Initial State**

**Gambar 3.23** *Start Point/Initial State*

#### 2. *Activity*(Aktivitas)

*Activity* menunjukkan aktivitas yang dilakukan atau yang sedang terjadi dalam *Activity* diagram.



**Gambar 3.24** *Activitiy*

### 3. *Action Flow*(Arah)

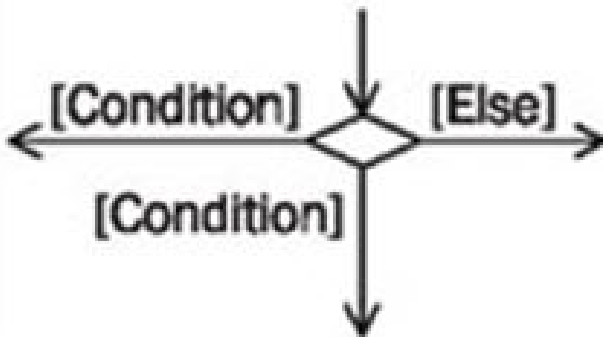
*Action Flow* digunakan untuk transisi dari suatu tindakan ke tindakan yang lain atau menunjukan aktivitas selanjutnya setelah aktivitas sebelumnya.



**Gambar 3.25** *Action Flow*

### 4. *Decision*(Keputusan)

*Decision* adalah suatu titik atau point pada activity diagram yang mengindikasikan suatu kondisi dimana ada kemungkinan perbedaan transisi.



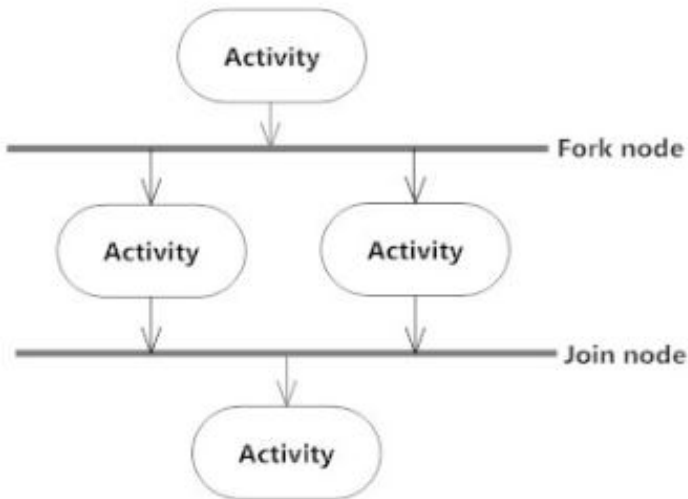
**Gambar 3.26** *Decision*

### 5. *Synchornization*

*Synchornization* dibagi menjadi 2 yaitu fork dan join.

- (a) Fork (percabangan) digunakan untuk memecah behaviour menjadi activity atau action yang paralel.
- (b) Join (penggabungan) untuk menggabungkan kembali activity atau action yang paralel.

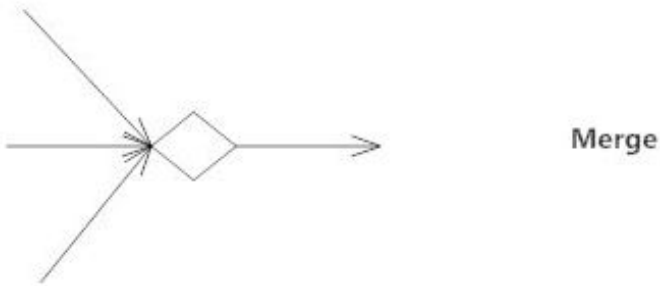
### Synchronization



**Gambar 3.27** *Synchornization*

## 6. Merge Event(Menggabungkan)

*Merge Event* berfungsi untuk menggabungkan flow yang dipecah oleh *decission*.



**Gambar 3.28** *Merge Event*

## 7. Swimlanes

*Swimlanes* berfungsi untuk memecah activity diagram menjadi baris dan kolom untuk membagi tanggung jawab obyek-obyek yang melakukan aktivitas.



**Gambar 3.29** *Swimlanes*

## 8. Final State/End Point(Titik Akhir)

*Final State/End Point* menunjukkan bagian akhir dari aktivitas.

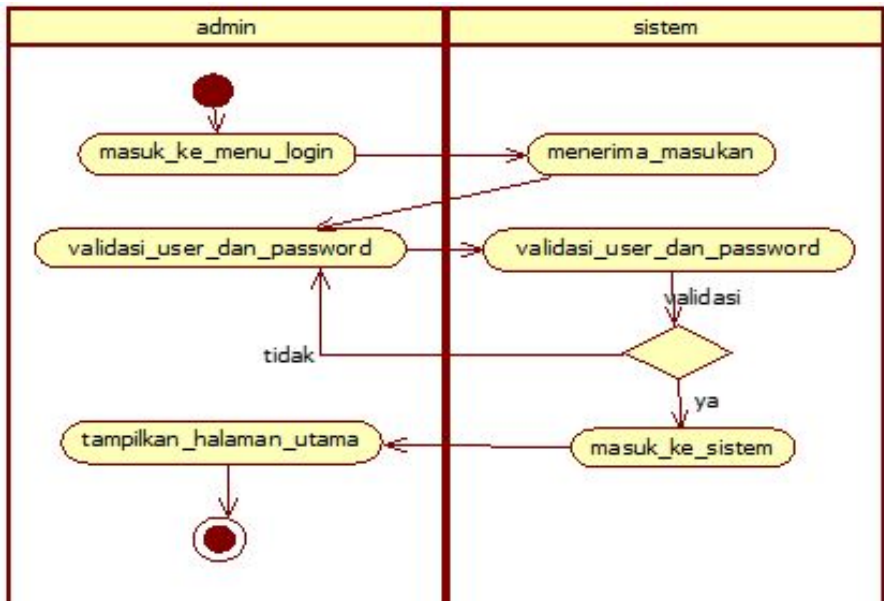


End Point Symbol

Gambar 3.30 Final State/End Point

maka dari itu di dalam buku ini di buatnya *Activity Diagram* seperti di bawah ini:

### 1. Activity diagram login



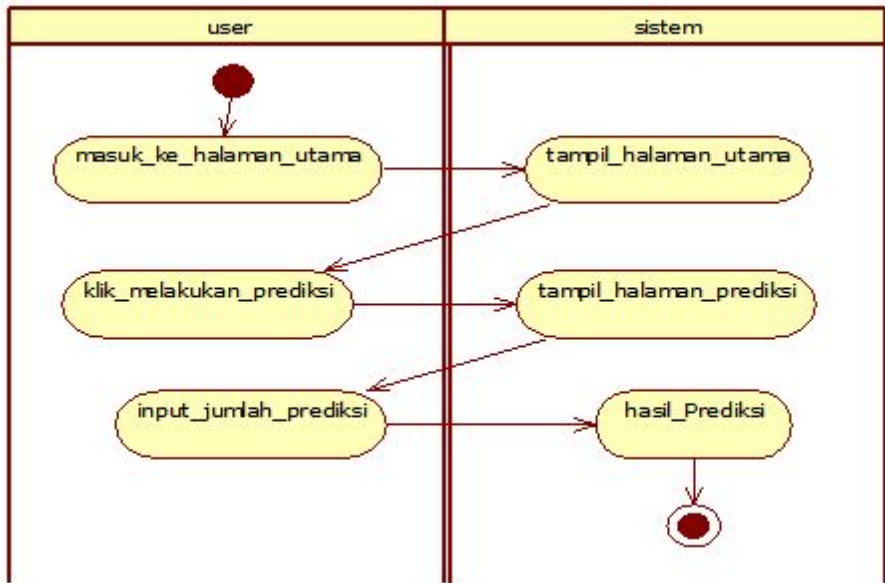
Gambar 3.31 Activity diagram login

Deskripsi *Activity diagram login* :

Proses login dimulai ketika admin memasukkan username dan password di form login akan melakukan passing terhadap username dan password yang telah di-inputkan kedalam koneksi database. Database kemudian akan melakukan validasi terhadap username dan password sesuai user yang terintegrasi didalam database. Setelah melakukan proses maka admin akan diarahkan ke halaman utama. Ketika admin melakukan logout, database akan memutuskan koneksi dari halaman utama.

### 2. Activity diagram Melakukan Input Prediksi



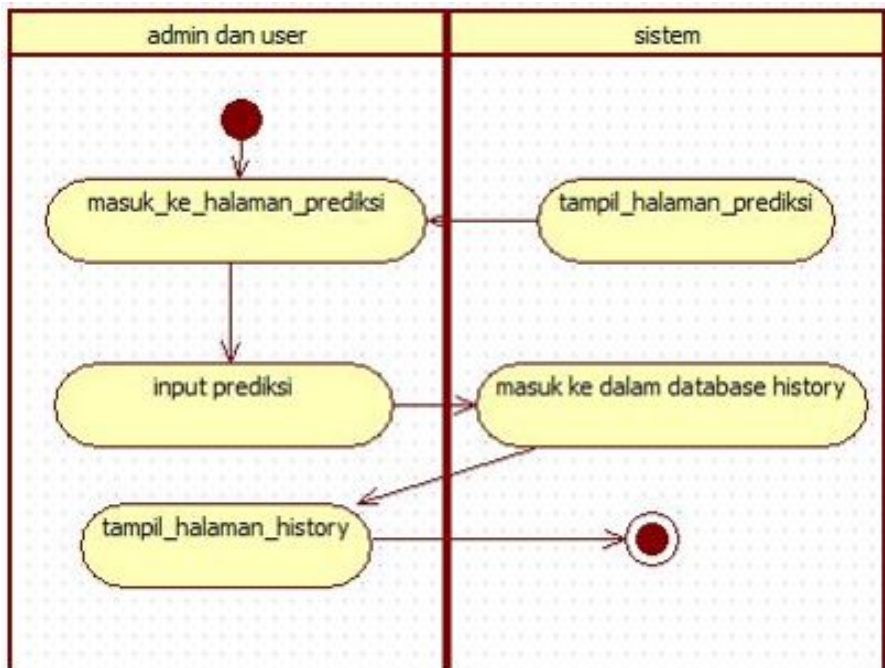


**Gambar 3.32** Activity diagramMelakukan Input Prediksi

Deskripsi Activity diagramMelakukan Input Prediksi :

Pada saat masuk ke halaman utama , adakan muncul klik untuk melakukan prediksi, setelah di klik akan masuk ke halaman prediksi di mana akan memasukkan jumlah hari yang akan di prediksi, setelah itu akan muncul jumlah prediksi yang keluar dari system.

### 3. Activity diagramCetak laporan hasil Prediksi :

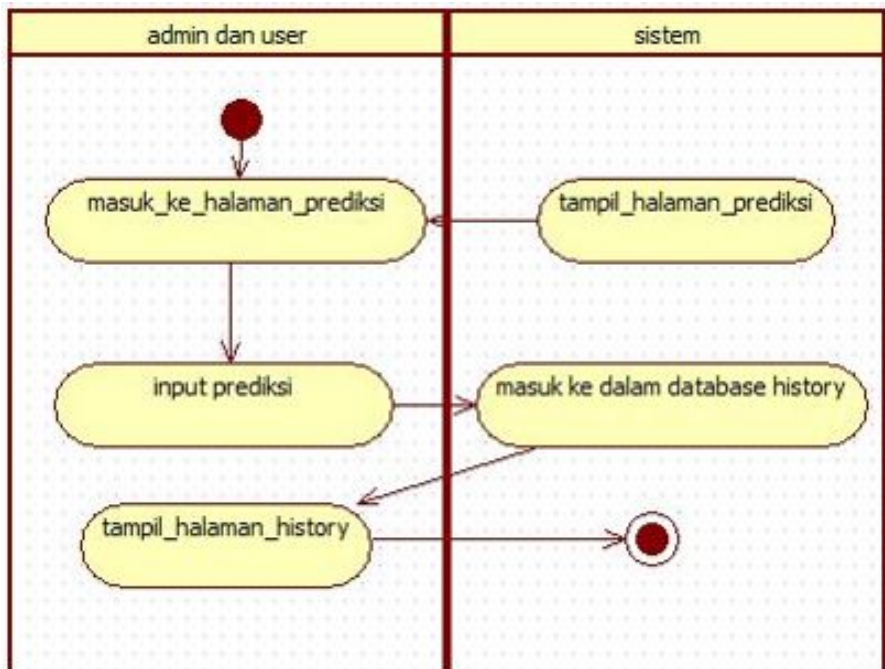


**Gambar 3.33** Activity diagram Cetak laporan hasil Prediksi :

Deskripsi Activity diagram Cetak laporan hasil Prediksi :

Proses cetak laporan hasil prediksi dimulai ketika user masuk ke halaman prediksi, setelah melakukan input prediksi akan tampil hasil dan dapat di import dalam bentuk PDF,

#### 4. Activity diagram History Prediksi

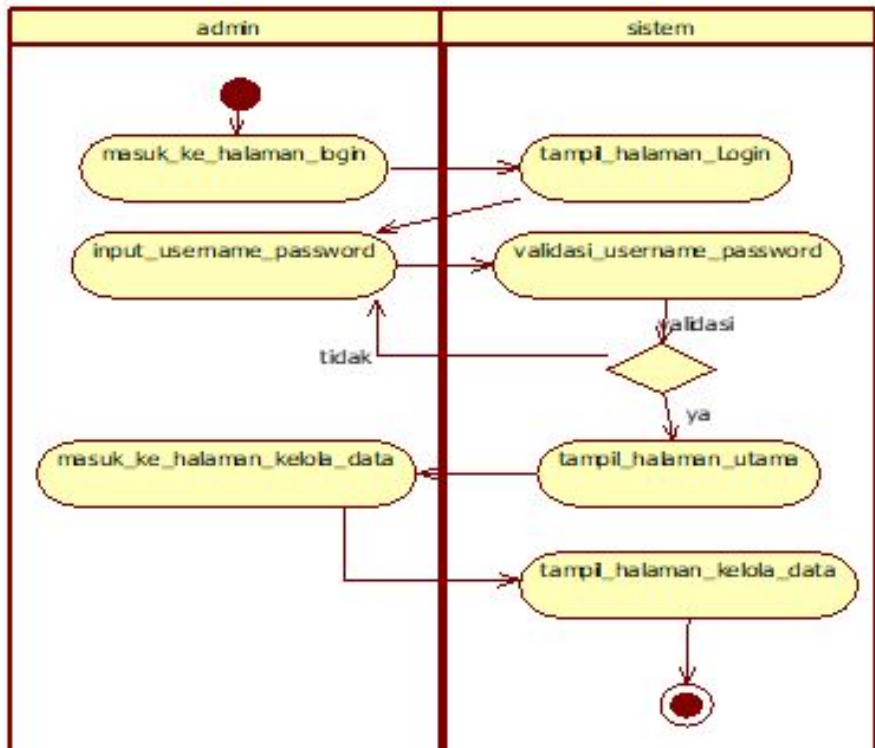


**Gambar 3.34** Activity diagram HistoryPrediksi

Deskripsi Activity diagram HistoryPrediksi :

Pada saat masuk ke halaman utama , lalu mengklik tombol prediksi, lalu akan masuk ke halaman prediksi, setelah melakukan prediksi , data prediksi akan masuk ke dalam history prediksi yang sewaktu waktu bisa di lihat kembali.

## 5. Activity diagram Kelola data



**Gambar 3.35** Activity diagramKelola data

Deskripsi Activity diagramKelola data :

Proses kelola data dimulai ketika admin memasukkan username dan password di form login, kemudian form login akan melakukan passing terhadap username dan password yang telah diinputkan kedalam koneksi database. Database kemudian akan melakukan validasi terhadap username dan password sesuai admin yang terintegrasi didalam database. Setelah melakukan proses validasi akan muncul konfirmasi tergantung validasinya benar atau tidak, jika valid atau benar maka admin akan diarahkan ke halaman utama. Setelah admin di halaman utama, admin dapat memilih halaman data prediksi untuk melakukan pengolahan data.

### 3.2.7 Perancangan Antara Muka

Desain Antarmuka Pengguna (bahasa Inggris: *User Interface Design*) atau rekayasa antarmuka pengguna adalah desain untuk komputer, peralatan, mesin, perangkat komunikasi mobile, aplikasi perangkat lunak, dan situs web yang berfokus pada pengalaman pengguna (bahasa Inggris: *User Experience*) dan interaksi.

Desain Antarmuka Pengguna yang baik dapat memberikan penyelesaian pekerjaan dengan menggunakan tangan tanpa menarik perhatian yang tidak perlu terhadap dirinya sendiri. Desain grafis dapat dimanfaatkan untuk mendukung kegunaan (bahasa Inggris: Usability). Proses desain haruslah seimbang antara fungsi teknis dan elemen visual (misalnya, model mental) untuk menciptakan sebuah sistem yang tidak hanya bisa beroperasi tetapi juga dapat digunakan dan disesuaikan dengan kebutuhan pengguna.

Desain Antarmuka Pengguna terlibat dalam berbagai proyek dari sistem komputer, untuk mobil, untuk pesawat komersial; semua proyek ini melibatkan banyak interaksi manusia dasar yang sama dan juga membutuhkan beberapa keterampilan yang unik dan pengetahuan. Akibatnya, desainer cenderung mengkhususkan diri pada jenis proyek tertentu dan memiliki kemampuan berpusat di sekitar keahlian mereka, apakah itu desain software, penelitian pengguna, desain web, atau desain industri.

Tujuan dari Desain Antarmuka Pengguna adalah untuk membuat interaksi pengguna sesederhana dan seefisien mungkin, dalam hal mencapai tujuan pengguna atau apa yang sering disebut dengan user-centered design. dan adapun tujuan lainnya seperti sebagai berikut :

1. Memudahkan Input Data Masukan Oleh User

User Interface yang baik adalah sebuah interface yang mudah dimengerti dan simple. User Interface harus memberikan sebuah kemudahan bagi user untuk dapat berinteraksi dengan program.

2. Membuat Program Menjadi Menarik

Tujuan yang kedua adalah dari sisi seni atau estetika dari program. Kita dapat melihat setiap aplikasi baik web, desktop maupun aplikasi mobile pastinya memiliki sebuah ciri khas interface tersendiri. Contoh seperti Facebook dengan variasi warna birunya, Line dengan warna hijau muda dengan icon icon nya yang khas. Kekhasan ini pastinya sangat berpengaruh pada ketertarikan orang terhadap suatu program.

3. Menampilkan hasil pemrosesan data oleh program

Selanjutnya adalah penampilan data, pernahkan anda menggunakan aplikasi tanpa user interface pada linux seperti aplikasi ifconfig atau aplikasi firewall tanpa user interface. Anda yang tidak terbiasa menggunakan terminal pasti akan kebingungan dan malas untuk membaca data dari terminal tanpa ada user interface yang canggih untuk menampilkan data dan berinteraksi dengan komputer. Dengan adanya user interface, data dapat ditampilkan dalam bentuk grafis sehingga user dapat melihat langsung data yang ada pada program.

4. Memudahkan pembacaan data yang dihasilkan oleh program

Tujuan lainnya selain penampilan data adalah memudahkan pembacaan data dari sebuah program. Tanpa adanya user interface yang baik, diperlukan waktu yang lama untuk mengurutkan, mencari, atau memfilter data yang dihasilkan

dari sebuah program. Dengan adanya user interface ini, tugas umum seperti mengurutkan data, mencari, atau melakukan filter data akan menjadi lebih mudah, karena telah di handle pekerjaannya oleh user interface. User hanya memilih menu pada user interface dan selanjutnya data akan diproses sesuai dengan kebutuhan user.

#### 5. Efisiensi kerja yang dilakukan oleh user

Dengan adanya user interface, kerja dari user akan lebih efisien dan tidak membutuhkan waktu ekstra untuk mengerjakan suatu tugas ekstra yang sebenarnya dapat dilakukan oleh user interface. contoh, sebuah user interface yang baik pada aplikasi statistik penjualan akan dapat langsung menghasilkan grafik dan juga kurva mengenai ramalan penjualan tanpa harus diproses datanya secara manual oleh user. Tentunya hal ini akan sangat berpengaruh besar pada efisiensi waktu dari seorang pekerja. Dimana dia dapat memanfaatkan waktunya untuk melakukan tugas lainnya yang lebih penting.

#### 6. Memperkecil kesalahan masukan data yang dilakukan oleh user

Sebuah user interface yang baik dapat mengurangi kesalahan masukan data yang dilakukan oleh user. Sebagai contoh, user interface pada aplikasi program facebook yang berupa update status, user interface tersebut membatasi masukannya berupa text, gambar, dan mungkin video. Apabila kita memasukkan data lainnya semisal data wmv atau semacamnya yang tidak disupport oleh interface, maka data tersebut tidak akan ditampilkan dan terdapat peringatan mengenai kesalahan yang dilakukan oleh user

Manfaat User Interface untuk menghubungkan atau penterjemah informasi antara pengguna dengan system operasi, sehingga computer dapat digunakan. Dengan demikian, user interface bisa juga diartikan sebagai mekanisme inter-relasi atau integrasi total dari perangkat keras dan lunak yang membentuk pengalaman bekompuser. Use interface dari sisi software bias berbentuk Graphical User Interface (GUI) atau Command Line Interfae (CLI), sedangkan dari sisi hardware bias berbentuk Aplle Desktop Bus (ADB), USB, dan fire wire.

Elemen-Elemen dalam perancangan antarmuka adalah

##### 1. Mendefinisikan konsep.

Mengumpulkan kebutuhan-kebutuhan pengguna dan mendefinisikan desain secara konseptual.

##### 2. Memvalidasi konsep.

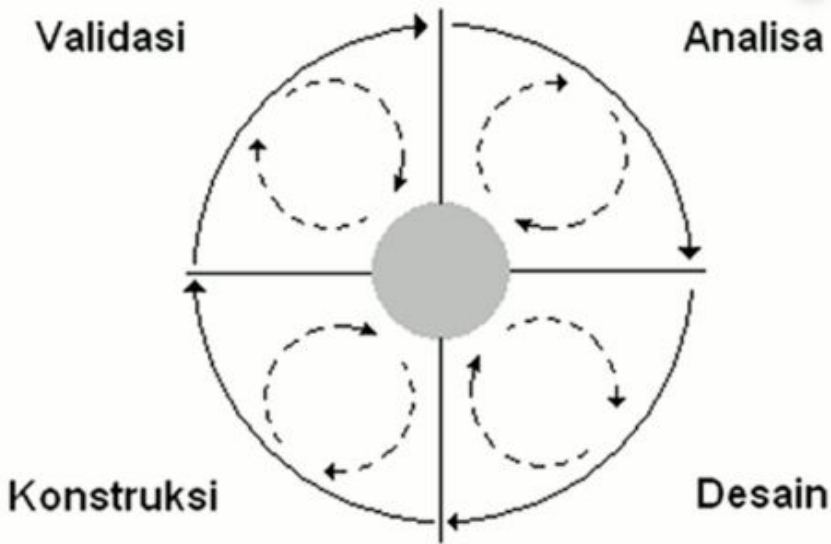
Mengevaluasi konseptual desain tersebut.

##### 3. Merancang.

Mengevaluasi prototype. Menandai dan memperbaiki masalah-masalah yang ditemukan.

##### 4. Pengembangan.

Melakukan pengujian secara berkala terhadap desain yang lebih dahulu dibuat dan desain yang paling terakhir dibuat. Menandai dan memperbaiki masalah-masalah yang ditemukan.



**Gambar 3.36** Empat Tahap Proses Perancangan Antarmuka

Proses yang secara rinci menggambarkan bagaimana perancangan dan pengembangan antarmuka terlihat pada gambar di atas. Empat tahap utama dalam proses tersebut adalah:

Mengumpulkan atau menganalisa informasi dari pengguna.

Merancang Antarmuka.

Mengembangkan Antarmuka.

Memvalidasi Antarmuka.

Proses-proses tersebut independen dari hardware dan software, sistem operasi dan peralatan yang digunakan untuk merancang dan mengembangkan produk. IBM Common User Access (CUA) interface design guide adalah yang pertama kali mendeskripsikan proses perancangan antarmuka secara iteratif.

#### 1. Mengumpulkan atau Menganalisa Informasi Pengguna.

Proses perancangan antarmuka dimulai dari memahami pengguna. Sebelum merancang antarmuka, kita harus mengetahui masalah apa yang ingin pengguna selesaikan dan bagaimana mereka melakukan pekerjaan mereka. Pengumpulan dan penganalisaan aktivitas-aktivitas pada tahap pertama ini dapat dijabarkan dalam lima langkah:

(a) Menentukan profil pengguna.

- (b) Melakukan analisa terhadap task-task pengguna.
- (c) Mengumpulkan kebutuhan-kebutuhan pengguna.
- (d) Menganalisa user environments.
- (e) Mencocokkan kebutuhan tersebut dengan task.

## 2. Merancang Antarmuka.

Dalam merancang antarmuka ada beberapa tahapan yang harus dilalui, yaitu:

- (a) Menjelaskan kegunaan dan tujuan.
- (b) Menetapkan icon objek, views, dan representasi visual.
- (c) Merancang objek dan jendela menu
- (d) Memperbaiki rancangan visual.

## 3. Mengembangkan Antarmuka.

Hal pertama yang bisa dilakukan dalam membangun antarmuka adalah membangun prototype. Membangun prototype adalah cara yang berharga dalam membuat rancangan awal dan membuat demonstrasi produk dan penting untuk pengujian kegunaan antarmuka. Dari prototype tersebut, perancang antarmuka dapat mulai membangun antarmuka secara utuh. Ketika membuat prototype, sangat penting untuk diingat bahwa prototype harus dapat di buang setelah digunakan (disposable). Jangan takut untuk membuang sebuah prototype. Tujuan dalam membuat prototype adalah untuk mempercepat dan mempermudah dalam memvisualisasikan desain alternatif dan konsep, bukan untuk membangun kode yang akan digunakan sebagai bagian dari produk.

## 4. Melakukan Validasi Terhadap Antarmuka.

Evaluasi kegunaan adalah bagian penting dari proses pengembangan, untuk mengetahui bagaimana tanggapan pengguna terhadap antarmuka yang telah dibuat. Evaluasi ini akan digunakan untuk memperbaiki kekurangan pada antarmuka yang telah dibangun. Aturan emas dalam perancangan antarmuka:

- (a) Buat Pengguna menguasai antarmuka.
- (b) Kurangi user's memory load
- (c) Buat antarmuka konsisten

Prinsip prinsip dalam merancang user interface

1. *User familiarity* / Mudah dikenali : gunakan istilah, konsep dan kebiasaan user bukan computer (misal: sistem perkantoran gunakan istilah letters, documents, folders bukan directories, file, identifiers. – jenis document open office.
2. *Consistency* / selalu begitu : Konsisten dalam operasi dan istilah di seluruh sistem sehingga tidak membingungkan. – layout menu di open office mirip dgn layout menu di MS office.



3. *Minimal surprise* / Tidak buat kaget user : Operasi bisa diduga prosesnya berdasarkan perintah yang disediakan.
4. *Recoverability*/pemulihan : Recoverability ada dua macam: Confirmation of destructive action (konfirmasi terhadap aksi yang merusak) dan ketersediaan fasilitas pembatalan (undo).
5. *User guidance* / bantuan : Sistem manual online, menu help, caption pada icon khusus tersedia.
6. *User diversity* /keberagaman : Fasilitas interaksi untuk tipe user yang berbeda disediakan. Misalnya ukuran huruf bisa diperbesar.

Ada 5 tipe utama interaksi untuk user interaction:

### 1. *manipulation*

Pengoperasian secara langsung: interaksi langsung dengan objek pada layar. Misalnya delete file dengan memasukkannya ke trash. Contoh: Video games.

Kelebihan: Waktu pembelajaran user sangat singkat, feedback langsung diberikan pada tiap aksi sehingga kesalahan terdeteksi dan diperbaiki dengan cepat.

Kekurangan : Interface tipe ini rumit dan memerlukan banyak fasilitas pada sistem komputer, cocok untuk penggambaran secara visual untuk satu operasi atau objek

### 2. *Menu selection*

Pilihan berbentuk menu: Memilih perintah dari daftar yang disediakan. Misalnya saat click kanan dan memilih aksi yang dikehendaki.

Kelebihan : User tidak perlu ingat nama perintah. Pengetikan minimal. Kesalahan rendah.

Kekurangan :Tidak ada logika AND atau OR. Perlu ada struktur menu jika banyak pilihan. Menu dianggap lambat oleh expert user dibanding command language.

### 3. *Form fill-in*

Pengisian form : Mengisi area-area pada form. Contoh: Stock control.

Kelebihan : Masukan data yang sederhana. Mudah dipelajari

Kekurangan : Memerlukan banyak tempat di layar. Harus menyesuaikan dengan form manual dan kebiasaan user.

### 4. *Command language*

Perintah tertulis: Menuliskan perintah yang sudah ditentukan pada program. Contoh: operating system.

Kelebihan : Perintah diketikan langsung pada system. Misal UNIX, DOS command. Bisa diterapkan pada terminal yang murah. Kombinasi perintah bisa dilakukan. Misal copy file dan rename nama file.

Kekurangan : Perintah harus dipelajari dan diingat cara penggunaannya tidak cocok untuk user biasa. Kesalahan pakai perintah sering terjadi. Perlu ada sistem pemulihan kesalahan. Kemampuan mengetik perlu.

## 5. *Natural language*

Perintah dengan bahasa alami: Gunakan bahasa alami untuk mendapatkan hasil. Contoh: search engine di Internet.

Kelebihan: Perintah dalam bentuk bahasa alami, dengan kosa kata yang terbatas (singkat) misalnya kata kunci yang kita tentukan untuk dicari oleh search engine. Ada kebebasan menggunakan kata-kata.

Kekurangan: Tidak semua sistem cocok gunakan ini. Jika digunakan maka akan memerlukan banyak pengetikan.

### *User Support*

User guidance meliputi semua fasilitas sistem untuk mendukung user termasuk on-line help, error messages, user manual. User guidance perlu disatukan dengan UI untuk bantu user saat membutuhkan informasi tentang sistem atau saat ada kesalahan. Help System dan sistem message (pesan kesalahan) adalah bentuk dari user guidance. Error Messages sangat penting, karena error message yang buruk cenderung ditolak oleh user dan error message sebaiknya berpedoman pada faktor-faktor pada Tabel dibawah.

Context	User guidance memberikan pesan sesuai konteks yang dialami oleh pengguna
Experience	Pesan yang panjang mengganggu bagi yang sudah biasa, tapi bagi pengguna baru pesan yang pendek membingungkan. User guidance perlu menjawab kebutuhan ini.
Skill level	Pesan dibuat sesuai dengan kemampuan pengguna dan pengalamannya.
Style	Harus bersifat positif. Lebih baik bersifat aktif dari pada pasif. Harus sopan, tidak menghina atau bergurau
Culture	Jika mungkin, pesan disesuaikan dengan budaya. Mungkin kata atau warna yang digunakan disesuaikan dengan budaya setempat.

**Gambar 3.37** *User Support*

Perancangan Input untuk menghasilkan informasi, diperlukan perancangan input sebagai dasar dalam perancangan suatu sistem, sehingga menghasilkan suatu output yang diharapkan. Perancangan yang terdapat dalam buku ini adalah :

### 1. Perancangan Antarmuka Halaman Login

The image shows a screenshot of a web application window titled "Nte Prediksi". The window has a standard Windows-style title bar with minimize, maximize, and close buttons. Inside the window, there is a navigation menu at the top with links: "Nte Prediski", "Home", "About", "Prediksi", and "Login". Below the menu is a horizontal bar containing the text "Home/Login". Underneath this bar is a large rectangular frame containing the login form. The form has a heading "Login" and two input fields: "Username" and "Password", each followed by a "Textbox" label. The "Textbox" labels are inside rectangular input boxes.

**Gambar 3.38** Perancangan Antarmuka Halaman Login

Keterangan:

Isikan Username dan Password kemudian klik Login. Apabila proses login berhasil maka akan masuk ke form halaman utama yang sesuai dengan hak akses yang dimiliki.

## 2. Perancangan Antarmuka Halaman Utama



**Gambar 3.39** Perancangan Antarmuka Halaman Utama

Keterangan:

User akan mulai pada halaman ini, User bisa langsung melakukan prediksi dengan mengklik button Prediksi , sehingga akan langsung ke halaman Prediksi.

### 3. Perancangan Antarmuka Halaman Prediksi

Nte Prediksi

Home About Prediksi Login

Home/Prediksi

[No]	Data Prediksi
[1]	Jumlah

Jumlah Hari Prediksi ▼

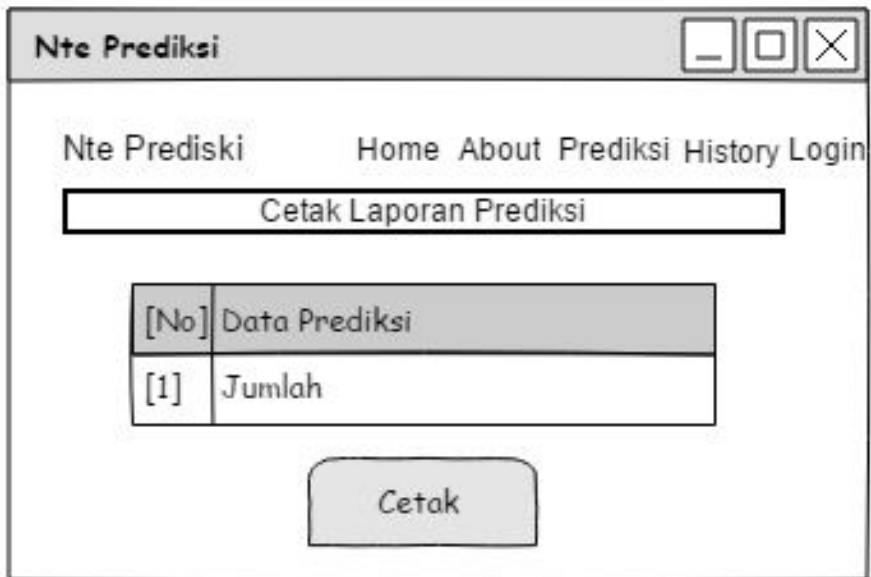
Hasil Prediksi 2 hari kedepan adalah ...

**Gambar 3.40** Perancangan Antarmuka Halaman Prediksi

Keterangan:

Halaman Prediksi, User akan memilih jumlah hari yang di prediksi seperti yang sudah di masukkan di ComboBox yang sudah di sediakan. Sehingga akan muncul hasil prediksi.

#### 4. Perancangan Antarmuka Halaman Cetak Hasil Laporan Prediksi

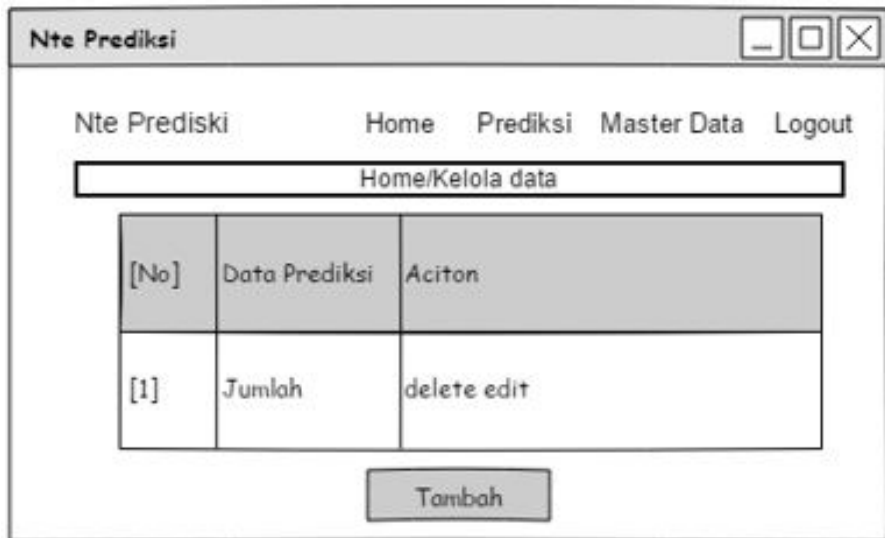


**Gambar 3.41** Perancangan Antarmuka Halaman Cetak Hasil Laporan Prediksi

Keterangan:

Halaman cetak apabila di klik tombol cetak akan masuk ke halaman baru yang berformat PDF yang bisa di download.

## 5. Perancangan Antarmuka Halaman Kelola Data

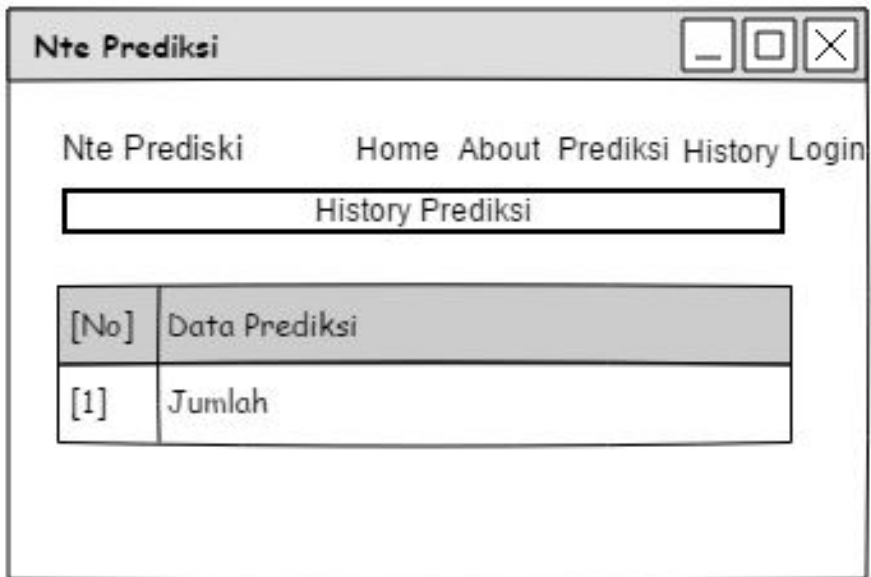


**Gambar 3.42** Perancangan Antarmuka Halaman Kelola Data

Keterangan:

Halaman Kelola Data yang hanya bisa dilakukan oleh admin, admin bisa menginput data menghapus data dan mengedit data.

## 6. Perancangan Antarmuka Halaman History



**Gambar 3.43** Perancangan Antarmuka Halaman History

Keterangan:

pada halaman history akan menampilkan halaman data prediksi yang sudah dilakukan , yang akan terimput otomatis setelah kita melakukan prediksi.





## BAB 4

---

# IMPLEMENTASI

---

### 4.1 Pembuatan Data Base

#### 4.1.1 Perancangan Database

Pada tulisan ini akan dibahas tahap-tahap dalam perancangan database. Pada database yang digunakan oleh single user atau hanya beberapa user saja, perancangan database tidak sulit.

Tetapi jika ukuran database yang sedang atau besar perancangan database akan menjadi sangat kompleks. Oleh karena itu para pemakai mengharapkan penggunaan database yang sedemikian rupa sehingga sistem harus dapat memenuhi kebutuhan-kebutuhan seluruh user tersebut. Tentunya agar kebutuhan-kebutuhan tersebut dapat terlayani dengan baik maka harus dilakukan perancangan database yang baik pula, aktifitas-aktifitas apa saja yang dilakukan pada tahap tersebut.

#### Tujuan Perancangan Database

1. Untuk memenuhi informasi yang berisikan kebutuhan-kebutuhan user secara khusus dan aplikasi aplikasinya.

2. Memudahkan pengertian struktur informasi.
3. Mendukung kebutuhan-kebutuhan pemrosesan dan beberapa obyek penampilan (*response time, processing time, dan storage space*).

### **Proses Perancangan Database**

1. Tahap 1, Pengumpulan data dan analisis
2. Tahap 2, Perancangan database secara konseptual
3. Tahap 3, Pemilihan DBMS
4. Tahap 4, Perancangan database secara logika (data model mapping)
5. Tahap 5, Perancangan database secara fisik
6. Tahap 6, Implementasi Sistem database

Secara khusus proses perancangan berisi 2 aktifitas paralel:

1. Aktifitas yang melibatkan perancangan dari isi data dan struktur database,
2. Aktifitas mengenai perancangan pemrosesan database dan aplikasi-aplikasi perangkat lunak.

Tahap 1 dan 6 kadang-kadang bukan merupakan bagian dari perancangan database. Sedangkan yang merupakan inti dari proses perancangan database adalah pada tahap 2, 4, 5.

#### **1. Tahap 1 Pengumpulan data dan analisa**

Merupakan suatu tahap dimana kita melakukan proses indentifikasi dan analisa kebutuhan-kebutuhan data dan ini disebut pengumpulan data dan analisa. Untuk menentukan kebutuhan-kebutuhan suatu sistem database, kita harus mengenal terlebih dahulu bagian-bagian lain dari sistem informasi yang akan berinteraksi dengan sistem database, termasuk para user yang ada dan para user yang baru beserta aplikasi-aplikasinya. Kebutuhan-kebutuhan dari para user dan aplikasi-aplikasi inilah yang kemudian dikumpulkan dan dianalisa. Berikut ini adalah aktifitas-aktifitas pengumpulan data dan analisa:

- (a) Menentukan kelompok pemakai dan bidang-bidang aplikasinya
- (b) Peninjauan dokumentasi yang ada
- (c) Analisa lingkungan operasi dan pemrosesan data
- (d) Daftar pertanyaan dan wawancara

#### **2. Tahap 2, Perancangan database secara konseptual**

Pada tahap ini akan dihasilkan *conceptual schema* untuk database yang tergantung pada sebuah DBMS yang spesifik. Sering menggunakan sebuah *high-level data model* seperti ER/EER model selama tahap ini. Dalam *conceptual schema*,

kita harus merinci aplikasi-aplikasi database yang diketahui dan transaksi-transaksi yang mungkin. Tahap perancangan database secara konseptual mempunyai 2 aktifitas paralel:

(a) Perancangan skema konseptual

Menguji kebutuhan-kebutuhan data dari suatu database yang merupakan hasil dari tahap 1 dan menghasilkan sebuah conceptual database schema pada DBMS-independent model data tingkat tinggi seperti EER (Enhanced Entity Relationship) model. Untuk menghasilkan skema tersebut dapat dihasilkan dengan penggabungan bermacam-macam kebutuhan user dan secara langsung membuat skema database atau dengan merancang skema-skema yang terpisah dari kebutuhan tiap-tiap user dan kemudian menggabungkan skema-skema tersebut. Model data yang digunakan pada perancangan skema konseptual adalah DBMS-independent dan langkah selanjutnya adalah memilih DBMS untuk melakukan rancangan tersebut.

(b) Perancangan transaksi

Menguji aplikasi-aplikasi database dimana kebutuhan-kebutuhannya telah dianalisa pada fase 1, dan menghasilkan perincian transaksi-transaksi ini. Kegunaan tahap ini yang diproses secara paralel bersama tahap perancangan skema konseptual adalah untuk merancang karakteristik dari transaksi-transaksi database yang telah diketahui pada suatu DBMS-independent. Transaksi-transaksi ini akan digunakan untuk memproses dan memanipulasi database suatu saat dimana database tersebut dilaksanakan.

### 3. Tahap 3, Pemilihan DBMS

Pemilihan database ditentukan oleh beberapa faktor diantaranya faktor teknik, ekonomi, dan politik organisasi. Contoh faktor teknik:

Keberadaan DBMS dalam menjalankan tugasnya seperti jenis-jenis DBMS (relational, network, hierarchical, dan lain-lain), struktur penyimpanan, dan jalur akses yang mendukung DBMS, pemakai, dan lain-lain. Faktor-faktor ekonomi dan organisasi yang mempengaruhi satu sama lain dalam pemilihan DBMS :

(a) Struktur data

Jika data yang disimpan dalam database mengikuti struktur hirarki, maka suatu jenis hirarki dari DBMS harus dipikirkan.

(b) Personal yang telah terbiasa dengan suatu sistem

Jika staf programmer dalam suatu organisasi sudah terbiasa dengan suatu DBMS, maka hal ini dapat mengurangi biaya latihan dan waktu belajar.

(c) Tersedianya layanan penjual

Keberadaan fasilitas pelayanan penjual sangat dibutuhkan untuk membantu memecahkan beberapa masalah sistem.

### 4. Tahap 4, Perancangan database secara logika (data model mapping)

Tahap selanjutnya adalah membuat sebuah skema konseptual dan skema eksternal pada model data dari DBMS yang terpilih. Tahap ini dilakukan oleh pemetaan skema konseptual dan skema eksternal yang dihasilkan pada tahap 2. Pada tahap ini, skema konseptual ditransformasikan dari model data tingkat tinggi yang digunakan pada tahap 2 ke dalam model data dari model data dari DBMS yang dipilih pada tahap 3. Pemetaan tersebut dapat diproses dalam 2 tingkat:

(a) Pemetaan system-independent

Pemetaan ke dalam model data DBMS dengan tidak mempertimbangkan karakteristik atau hal-hal yang khusus yang berlaku pada implementasi DBMS dari model data tersebut.

(b) Penyesuaian skema ke DBMS yang spesifik

Mengatur skema yang dihasilkan pada langkah 1 untuk disesuaikan pada implementasi yang khusus di masa yang akan datang dari suatu model data yang digunakan pada DBMS yang dipilih. Hasil dari tahap ini memakai perintah-perintah DDL (Data Definition Language) dalam bahasa DBMS yang dipilih yang menentukan tingkat skema konseptual dan eksternal dari sistem database. Tetapi 10 dalam beberapa hal, perintah-perintah DDL memasukkan parameter-parameter rancangan fisik sehingga DDL yang lengkap harus menunggu sampai tahap perancangan database secara fisik telah lengkap. Tahap ini dapat dimulai setelah pemilihan sebuah implementasi model data sambil menunggu DBMS yang spesifik yang akan dipilih. Contoh: jika memutuskan untuk menggunakan beberapa relational DBMS tetapi belum memutuskan suatu relasi yang utama. Rancangan dari skema eksternal untuk aplikasi-aplikasi yang spesifik seringkali sudah selesai selama proses ini.

## 5. Tahap 5, Perancangan database secara fisik

Perancangan database secara fisik merupakan proses pemilihan struktur-struktur penyimpanan dan jalur-jalur akses pada file-file database untuk mencapai penampilan yang terbaik pada bermacam-macam aplikasi. Selama fase ini, dirancang spesifikasi-spesifikasi untuk database yang disimpan yang berhubungan dengan struktur-struktur penyimpanan fisik, penempatan record dan jalur akses. Berhubungan dengan internal schema (pada istilah 3 level arsitektur DBMS). Beberapa petunjuk dalam pemilihan perancangan database secara fisik :

(a) Response time

Waktu yang telah berlalu dari suatu transaksi database yang diajukan untuk menjalankan suatu tanggapan. Pengaruh utama pada response time adalah di bawah pengawasan DBMS yaitu : waktu akses database untuk data item yang ditunjuk oleh suatu transaksi. Response time juga dipengaruhi oleh beberapa faktor yang tidak berada di bawah pengawasan DBMS, seperti penjadwalan sistem operasi atau penundaan komunikasi.

(b) Space utility

Jumlah ruang penyimpanan yang digunakan oleh file-file database dan struktur-struktur jalur akses.

(c) Transaction throughput

Rata-rata jumlah transaksi yang dapat diproses per menit oleh sistem database, dan merupakan parameter kritis dari sistem transaksi (misal : digunakan pada pemesanan tempat di pesawat, bank, dll). Hasil dari fase ini adalah penentuan awal dari struktur penyimpanan dan jalur akses untuk file-file database.

## 6. Tahap 6, Implementasi Sistem database

Setelah perancangan secara logika dan secara fisik lengkap, kita dapat melaksanakan sistem database. Perintah-perintah dalam DDL dan SDL(Storage Definition Language) dari DBMS yang dipilih, dihimpun dan digunakan untuk membuat skema database dan file-file database (yang kosong). Sekarang data basetersebut dimuat (disatukan) dengan datanya. Jika data harus dirubah dari sistem komputer sebelumnya, perubahan-perubahan yang rutin mungkin diperlukan untuk format ulang datanya yang kemudian dimasukkan ke database yang baru. Transaksi-transaksi database sekarang harus dilaksanakan oleh para programmer aplikasi. Spesifikasi secara konseptual diuji dan dihubungkan dengan kode program dengan perintah-perintah dari embedded DML yang telah ditulis dan diuji. Suatu saat transaksi-transaksi tersebut telah siap dan data telah dimasukkan ke dalam database, maka tahap perancangan dan implementasi telah selesai, dan kemudian tahap operasional dari sistem database dimulai.



## DAFTAR PUSTAKA

---



