

BUKU PANDUAN

BUKU PANDUAN



Kreatif Industri Nusantara

Penulis:

Cahyo Prianto S.Pd., M.T.
Sulpadianti Bunyamin

ISBN : xxxxxxxxxxxx

Editor:

M. Yusril Helmi Setyawan

Penyunting:

Syafrial Fachrie Pane
Khaera Tunnis
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. Sariasisih No. 54
Bandung 40151
Email : irc@poltekpos.ac.id

Cetakan Pertama, 2020

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

*'Jika Kamu tidak dapat
menahan lelahnya
belajar, Maka kamu harus
sanggup menahan
perihnya Kebodohan.'*

Imam Syafi'i

CONTRIBUTORS

SULPADIANTI BUNYAMIN, Informatics Engineering Student., Politeknik Pos Indonesia, Bandung, Indonesia

CONTENTS IN BRIEF

1	Codeigniter	1
2	<i>Codeigniter</i>	3
3	Pembangunan Aplikasi	7
4	Penjabaran Aplikasi	81
5	PENGKAJIAN DAN EVALUASI	199

DAFTAR ISI

Daftar Gambar	xiii
Daftar Tabel	xxiii
Foreword	xxvii
Kata Pengantar	xxix
Acknowledgments	xxxi
Acronyms	xxxiii
Glossary	xxxv
List of Symbols	xxxvii
Introduction	xxxix
<i>Sulpadianti Bunyamin</i>	
1 Codeigniter	1
2 Codeigniter	3
2.1 Sejarah <i>Codeigniter</i>	3
2.2 Perbedaan <i>Codeigniter V.2</i> dan <i>Codeigniter V.3</i>	5

3 Pembangunan Aplikasi	7
3.1 Pengenalan Awal Pembuatan Aplikasi	7
3.1.1 Latar Belakang Pembuatan Aplikasi	8
3.1.2 Identifikasi Masalah Aplikasi	8
3.1.3 Tujuan Aplikasi	8
3.1.4 Manfaat Aplikasi	8
3.1.5 Ruang Lingkup Pembangunan Aplikasi	9
3.1.6 Kebutuhan Aplikasi	9
3.2 Pembangunan Awal Aplikasi	13
3.2.1 Instalasi Software Pembangun Aplikasi	14
3.2.2 Panduan Penggunaan Awal Software Pendukung Aplikasi	41
4 Penjabaran Aplikasi	81
4.1 Data Mining	81
4.1.1 Tahapan dalam <i>Data Mining</i>	82
4.1.2 Teknik-Teknik Data mining	83
4.1.3 Sepuluh <i>Top Algoritma Data Mining</i>	88
4.1.4 Istilah dalam Data Mining	107
4.1.5 Fungsi Data Mining	108
4.1.6 Tujuan Data Mining	109
4.1.7 Contoh Penerapan Data Mining	110
4.1.8 Arsitektur Sistem Data mining	111
4.1.9 Software Aplikasi Data Mining	114
4.2 <i>Clustering</i>	118
4.2.1 Definisi	118
4.2.2 Macam- macam <i>Clustering</i>	118
4.2.3 Pendekatan Clustering	119
4.2.4 Hal-hal Terkait Dengan Clustering	120
4.2.5 Tahap validasi Cluster	121
4.2.6 <i>K-Means Clustering</i>	127
4.2.7 Pengelompokan untuk Pemahaman (<i>Understanding</i>)	128
4.2.8 Pengelompokan untuk <i>Utility</i>	128
4.2.9 Variasi	129
4.2.10 Pengaduan	170
4.2.11 Telkomsel	170
4.2.12 Jenis- jenis gangguan pada Jaringan Telkomsel	171
4.3 Analisis	171
4.3.1 Analisis Sistem Yang Sedang Berjalan	172

4.3.2	Analisis Sistem Yang Akan Dibangun	173
4.3.3	Analisis Perangkat yang Digunakan	174
4.4	Perancangan	175
4.4.1	<i>Use Case Diagram</i>	175
4.4.2	Definisi <i>Use Case</i>	176
4.4.3	Skenario <i>use case</i>	176
4.4.4	Skenario <i>Login</i>	177
4.4.5	Skenario <i>Validasi</i>	178
4.4.6	Skenario <i>Logout</i>	179
4.4.7	<i>Class Diagram</i>	179
4.4.8	<i>Sequence Diagram Login</i>	181
4.4.9	<i>Sequence Diagram Upload File</i>	182
4.4.10	<i>Sequence diagram Lihat Hasil Clustering</i>	183
4.4.11	<i>Activity diagram Login</i>	184
4.4.12	<i>Activity diagram Upload File</i>	185
4.4.13	<i>Activity diagram Lihat Hasil Clustering</i>	186
4.5	Perancangan Antarmuka	187
4.5.1	Perancangan Antarmuka Halaman Login	187
4.5.2	Perancangan Antarmuka Halaman Beranda	188
4.5.3	Perancangan Antarmuka Halaman Daftar User	189
4.5.4	Perancangan Antarmuka Halaman Daftar Gangguan	190
4.5.5	Perancangan Antarmuka Halaman Iterasi Data	191
4.6	<i>Interface</i>	192
4.6.1	<i>User Interface Login</i>	192
4.6.2	<i>User Interface Beranda</i>	193
4.6.3	<i>User Interface Daftar User</i>	194
4.6.4	<i>User Interface Jenis Gangguan</i>	195
4.6.5	<i>User Interface Iterasi Kmeans Clustering</i>	196
4.7	Pengujian <i>Blackbox</i>	196

5 PENGKAJIAN DAN EVALUASI 199

5.1	Penerapan Algoritma K-Means Clustering	199
5.1.1	Transformasi Data	199
5.1.2	Pengolahan Data	201
5.1.3	Hasil Clustering	201

Daftar Pustaka	203
----------------	-----

DAFTAR GAMBAR

2.1	Logo Ellis Lab	4
3.1	Diagram Alur Metodologi Penelitian	10
3.2	<i>Code Metode Elbow</i>	11
3.3	Hasil dari <i>Metode Elbow</i>	12
3.4	Perangkat Keras	13
3.5	Perangkat Lunak	13
3.6	Tutorial Instalasi XAMPP 01	14
3.7	Tutorial Instalasi XAMPP 02	15
3.8	Tutorial Instalasi XAMPP 03	16
3.9	Tutorial Instalasi XAMPP 04	16
3.10	Tutorial Instalasi XAMPP 05	17
3.11	Tutorial Instalasi XAMPP 06	18
3.12	Tutorial Instalasi XAMPP 07	19

3.13	Tutorial Instalasi XAMPP 08	20
3.14	Tutorial Instalasi XAMPP 09	21
3.15	Tutorial Instalasi XAMPP 10	22
3.16	Tutorial Instalasi XAMPP 11	23
3.17	Tutorial Instalasi XAMPP 12	24
3.18	Tutorial Instalasi Text Editor 01	25
3.19	Tutorial Instalasi Text Editor 02	26
3.20	Tutorial Instalasi Text Editor 03	27
3.21	Tutorial Instalasi Text Editor 04	27
3.22	Tutorial Instalasi Text Editor 05	28
3.23	Tutorial Instalasi Text Editor 06	29
3.24	Tutorial Instalasi Text Editor 07	30
3.25	Tutorial Instalasi Text Editor 08	31
3.26	Tutorial Instalasi Text Editor 09	32
3.27	Tutorial Instalasi Text Editor 10	32
3.28	Tutorial Instalasi Codeigniter 01	33
3.29	Tutorial Instalasi Codeigniter 02	33
3.30	Tutorial Instalasi Codeigniter 03	34
3.31	Tutorial Instalasi Codeigniter 04	34
3.32	Tutorial Instalasi Codeigniter 05	35
3.33	Tutorial Instalasi Bootstrap 01	36
3.34	Tutorial Instalasi Bootstrap 02	36
3.35	Tutorial Instalasi Bootstrap 03	37
3.36	Tutorial Instalasi Bootstrap 04	37
3.37	Tutorial Instalasi Bootstrap 05	38
3.38	Tutorial Instalasi Bootstrap 06	38
3.39	Tutorial Instalasi Bootstrap 07	39
3.40	Tutorial Instalasi Bootstrap 08	39

3.41	Tutorial Instalasi Bootstrap 09	39
3.42	Tutorial Instalasi Bootstrap 10	40
3.43	Tutorial Instalasi Bootstrap 11	40
3.44	Tutorial Instalasi Bootstrap 12	40
3.45	Tutorial Instalasi Bootstrap 13	41
3.46	Penggunaan XAMPP 01	42
3.47	Penggunaan XAMPP 02	43
3.48	Penggunaan XAMPP 03	43
3.49	Penggunaan XAMPP 04	44
3.50	Penggunaan XAMPP 05	45
3.51	Penggunaan XAMPP 06	46
3.52	Penggunaan XAMPP 07	47
3.53	Penggunaan XAMPP 08	47
3.54	Penggunaan XAMPP 09	48
3.55	Penggunaan XAMPP 10	49
3.56	Penggunaan XAMPP 11	50
3.57	Penggunaan XAMPP 12	51
3.58	Penggunaan XAMPP 13	52
3.59	Penggunaan XAMPP 14	53
3.60	Penggunaan XAMPP 15	53
3.61	Penggunaan XAMPP 16	54
3.62	Penggunaan XAMPP 17	55
3.63	Penggunaan XAMPP 18	56
3.64	Penggunaan XAMPP 19	56
3.65	Penggunaan XAMPP 20	57
3.66	Penggunaan XAMPP 21	58
3.67	Penggunaan XAMPP 22	58
3.68	Penggunaan XAMPP 23	59

3.69	Penggunaan XAMPP 24	60
3.70	Penggunaan XAMPP 25	61
3.71	Penggunaan XAMPP 26	62
3.72	Penggunaan XAMPP 27	62
3.73	Penggunaan XAMPP 28	63
3.74	Penggunaan Visual Studio Code 01	65
3.75	Penggunaan Visual Studio Code 02	66
3.76	Penggunaan Visual Studio Code 03	67
3.77	Penggunaan Visual Studio Code 04	68
3.78	Penggunaan Visual Studio Code 05	69
3.79	Penggunaan Visual Studio Code 06	70
3.80	Penggunaan Visual Studio Code 07	71
3.81	Penggunaan Visual Studio Code 08	72
3.82	Penggunaan Visual Studio Code 09	73
3.83	Penggunaan Visual Studio Code 10	74
3.84	Penggunaan Visual Studio Code 11	74
3.85	Penggunaan Visual Studio Code 12	76
3.86	Penggunaan Visual Studio Code 13	77
3.87	Penggunaan Visual Studio Code 14	78
3.88	Penggunaan Visual Studio Code 15	78
3.89	Penggunaan Visual Studio Code 16	79
4.1	Tahapan Data Mining	82
4.2	Teknik Data Mining	84
4.3	Decision Tree	87
4.4	Contoh C45	89
4.5	Perhitungan Gain	90
4.6	Perhitungan Entropi	91
4.7	Perhitungan	92

4.8	Perhitungan Entropi Total	92
4.9	Rumus Entropi Total	92
4.10	Menghitung gain	93
4.11	Ilustrasi SVM.	94
4.12	Ilustrasi SVM.	95
4.13	Ilustrasi SVM.	96
4.14	Pseudocode dari algoritma Apriori	99
4.15	pseudocode dari pembentukan kandidat itemset	99
4.16	Contoh Algoritma Apriori	100
4.17	Contoh Perhitungan Bayes	102
4.18	rumus bayes	102
4.19	rumus bayes	103
4.20	rumus bayes	103
4.21	rumus bayes	104
4.22	rumus bayes	104
4.23	Densitas Gauss	105
4.24	Rumus Mean	105
4.25	Rumus Deviasi Standar	106
4.26	Aturan pemisahan	107
4.27	Evaluasi Pola Data Mining	112
4.28	Proses Instalasi	116
4.29	Proses Import	116
4.30	Proses Pemilihan Paket Install	117
4.31		122
4.32	Output Deskriptive Statistics Analisi Cluster SPSS	123
4.33	Analisis Cluster Hirarki dengan SPSS	124
4.34	Agglomeration Schedule	125

4.35	Dendogram	126
4.36	Method Cluster	126
4.37	Membership Cluster	127
4.38	Contoh Data yang Digunakan	130
4.39	Contoh Input Data	131
4.40	Memilih Variabel Numerik dan Menghitung Statistika Deskriptif	131
4.41	Hasil Statistika Deskriptif	132
4.42	Menghitung Jarak Antar Objek	132
4.43	Jarak antar Data	133
4.44	Hasil Plot	134
4.45	Membuat Klasterisasi 4 Kelompok	135
4.46	Cluster	135
4.47	Membuat Data Frame Hasil Klaster	136
4.48	Membandingkan Hasil Klaster	137
4.49	Hasil Klaster	138
4.50	Menentukan Jumlah Kelompok Klaster yang Optimal	139
4.51	Plot Optimal Number of Cluster	139
4.52	Plot Optimal Number of Cluster	140
4.53	Input Libraries	141
4.54	Import Depedensi	141
4.55	Import Depedensi	142
4.56	Variabel NMHC dan RH	142
4.57	Variabel NMHC dan RH	143
4.58	NMHC dan RH yang sudah bersifat numerik	144
4.59	Visualisasi Data	144
4.60	Visualisasi	145
4.61	Visualisasi	145
4.62	Visualisasi	146

4.63	Array	146
4.64	Array	147
4.65	Konfigurasi Fungsi Kmeans Cluster	147
4.66	Penentuan Nilai K	148
4.67	Pusat Cluster	148
4.68	Tambah Kolom	149
4.69	Visualisasi Hasil Cluster	149
4.70	Visualisasi Hasil Cluster	150
4.71	GUI yang disediakan WEKA	151
4.72	Ketika weather.arff sudah dimasukkan	151
4.73	Memilih SImpleKMeans	152
4.74	Tampilan SimpleKMeans sudah siap untuk mulai	152
4.75	Setingan tambahan jika diinginkan	153
4.76	Hasil Clusternya	153
4.77	Cluster yang terbentuk per instance nya	154
4.78	Tampilan Download Rapidminer	154
4.79	Install Rapidminer	155
4.80	Halaman Instalasi	156
4.81	Proses menjalankan RapidMiner	157
4.82	Pengisian Data	158
4.83	Proses <i>Kmeans</i>	159
4.84	<i>Text Processing</i>	160
4.85	<i>Create Document</i>	161
4.86	<i>Edit Text</i>	161
4.87	<i>Pengisian Dokumen</i>	162
4.88	<i>Procces Document</i>	162
4.89	<i>Procces Document</i>	163
4.90	<i>Isikan Operator Tokenize</i>	163

4.91	<i>Operator Stopword</i>	164
4.92	<i>Penggunaan Notepad sebagai Database</i>	164
4.93	<i>masukan file pada parameter stopword</i>	165
4.94	<i>masukan operator stem</i>	165
4.95	<i>File Stem</i>	166
4.96	<i>File Stem</i>	166
4.97	<i>File Stem</i>	167
4.98	<i>Parameter K-means Clustering</i>	167
4.99	<i>Maxruns</i>	168
4.100	<i>Proses Cluster</i>	168
4.101	<i>Cluster Model dan ExampleSet</i>	169
4.102	<i>ExampleSet</i>	169
4.103		172
4.104		173
4.105	<i>Spesifikasi Perangkat keras</i>	174
4.106	<i>Spesifikasi Perangkat Lunak</i>	175
4.107	<i>Usecase Diagram</i>	175
4.108	<i>Definisi use case</i>	176
4.109	<i>Skenario use case</i>	176
4.110	<i>Skenario Login</i>	177
4.111	<i>Skenario Validasi</i>	178
4.112	<i>Skenario Log out</i>	179
4.113	<i>Class Diagram</i>	180
4.114	<i>Sequence Diagram Login</i>	181
4.115	<i>Sequence Diagram Upload File</i>	182
4.116	<i>Sequence Diagram Melihat Hasil Clustering</i>	183

4.117	<i>Activity Diagram Login</i>	184
4.118	<i>Activity Diagram Upload File</i>	185
4.119	<i>Activity Diagram Lihat Hasil Cluster</i>	186
4.120	<i>Halaman Login</i>	187
4.121	<i>Halaman Beranda</i>	188
4.122	<i>Halaman Daftar User</i>	189
4.123	<i>Halaman Gangguan</i>	190
4.124	<i>Halaman Iterasi Data</i>	191
4.125	<i>Halaman Login</i>	192
4.126	<i>Halaman Beranda</i>	193
4.127	<i>Halaman Daftar User</i>	194
4.128	<i>Halaman Jenis Gangguan</i>	195
4.129	<i>Halaman Iterasi Kmeans Clustering</i>	196
4.130	Pengujian <i>Blackbox</i>	197
5.1	Inisialisasi Wilayah	200
5.2	Inisialisasi Gangguan	200
5.3	Inisialisasi Durasi	201

DAFTAR TABEL

2.1	perbedaan versi codeigniter	5
-----	-----------------------------	---

Listings

FOREWORD

Sepatah kata dari Kaprodi, Kabag Kemahasiswaan dan Mahasiswa

KATA PENGANTAR

Buku ini diciptakan bagi para pembaca yang ingin mengetahui langkah-langkah dalam membangun sebuah program aplikasi.

SULPADIANTI BUNYAMMIN

Bandung, Jawa Barat

Januari, 2020

ACKNOWLEDGMENTS

Terima kasih atas semua pihak yang telah memberikan masukan dan juga bantuan sehingga memudahkan penulis dalam membuat buku ini.

Terima kasih ini juga ditujukan khusus untuk pembimbing internal penulis dan seluruh dosen program studi D4 Teknik Informatika yang memberi jalan dan saran dalam membuat buku ini.

F.

ACRONYMS

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

GLOSSARY

git	Merupakan manajemen sumber kode yang dibuat oleh linus torvald.
bash	Merupakan bahasa sistem operasi berbasiskan *NIX.
linux	Sistem operasi berbasis sumber kode terbuka yang dibuat oleh Linus Torvald

SYMBOLS

A Amplitude

& Propositional logic symbol

a Filter Coefficient

B Number of Beats

INTRODUCTION

SULPADIANTI BUNYAMIN

Student

Bandung, Jawa Barat, Indonesia

Pada era disruptif saat ini. git merupakan sebuah kebutuhan dalam sebuah organisasi pengembangan perangkat lunak. Buku ini diharapkan bisa menjadi penghantar para programmer, analis, IT Operation dan Project Manajer. Dalam melakukan implementasi git pada diri dan organisasinya.

Rumusnya cuman sebagai contoh aja biar keren[?].

$$ABC\mathcal{DEF}\alpha\beta\Gamma\Delta \sum_{def}^{abc} \quad (I.1)$$

BAB 1

CODEIGNITER

BAB 2

CODEIGNITER

2.1 Sejarah *Codeigniter*

Sejarah codeigniter yaitu pertama kali dirilis tahun 28 Februari 2006 ditulis oleh EllisLab versi stabil 2.1.0. Kemudian diambil alih dan dikembangkan British Columbia Institute of Technology (BCIT) yang merupakan Sekolah Tinggi Teknologi di Kanada pada bulan Oktober 2014 sampai dengan sekarang *web application framework* yang bersifat *open source* dimana digunakan untuk membangun aplikasi php dinamis [1].



Gambar 2.1 Logo Ellis Lab

Setelah perkembangan *Codeigniter* dibawah Ellis Lab, pada bulan Oktober 2014 CI secara resmi diambil alih dan dikembangkan oleh **British Columbia Institute of Technology** (BCIT) yang merupakan Sekolah Tinggi Teknologi di Kanada, dan pengembangan tersebut masih berlanjut sampai sekarang dengan versi 3.1.9. *Institut Teknologi British Columbia* adalah lembaga politeknik publik di Burnaby, British Columbia. BCIT pertama kali didirikan sebagai Sekolah Kejuruan British Columbia pada tahun 1960. Sejak didirikan, lembaga ini telah menampung lebih dari 125.000 alumni. Institute ini beroperasi sebagai sekolah kejuruan dan teknis, menawarkan magang untuk perdagangan terampil dan diploma dan gelar dalam pendidikan kejuruan untuk teknisi dan pekerja terampil dalam profesi seperti teknik, akuntansi, administrasi bisnis, komunikasi siaran / media, seni digital, keperawatan, kedokteran, arsitektur, dan hukum.

Berdasarkan pengembangannya, tujuan utama dari *Codeigniter* ialah untuk membantu developer dalam mengerjakan aplikasi lebih cepat dan mudah. *Codeigniter* memberikan berbagai macam library untuk mempermudah dalam pengembangan. *Codeigniter* dibangun menggunakan konsep *Model-View-Controller* (MVC) development pattern. MVC merupakan salah satu arsitektur aplikasi yang memisahkan antarmuka/tampilan (user interface), data, dan proses sehingga memungkinkan untuk melakukan pengembangan atau pemeliharaan aplikasi secara lebih efektif dan efisien. Dalam *Codeigniter*, browser berinteraksi melalui controller. Controller akan menerima dan membalas semua permintaan dari browser. Ketika controller membutuhkan data, maka controller akan meminta ke model. Sedangkan untuk tampilan pada user ditangani oleh view. Jadi otak dari aplikasi ada di controller, maka aplikasi ada di viewed, data berada di model [2].

2.2 Perbedaan *Codeigniter V.2* dan *Codeigniter V.3*

Ada beberapa perbedaan pada kedua versi berikut dimana dapat dipahami melalui tabel dibawah ini:

No.	Versi 2	Versi 3
1	Berlisensi Open Source	Berlisensi MIT
2	Database drivernya Mysql	Database drivernya sudah dapat dialihkan ke Mysqli
3	Minimum PHP versi 5.1.6	Minimum PHP versi 5.3.7
4	User Agent Yang Terbatas	User Agent yang mempunyai seperti iOS, windows dll

Tabel 2.1 perbedaan versi codeigniter

BAB 3

PEMBANGUNAN APLIKASI

3.1 Pengenalan Awal Pembuatan Aplikasi

Pada pembangunan aplikasi tentunya memiliki dasar dan juga tujuan pembuatan. Pembahasan kali ini akan mengarah kepada latar belakang maupun penjelasan awal terkait pembangunan aplikasi. Masalah yang muncul dan juga tujuan pencapaian akan dijabarkan sehingga dalam proses pembangunan aplikasi yang akan dilakukan anda dapat memahami maksud dari aplikasi ini. Pada pembangunan sebuah aplikasi sangat membutuhkan adanya pemahaman awal sehingga apa yang dikerjakan dalam lebih terstruktur, efektif dan sesuai tujuan pembuatan. Pembuatan program didasari akan pemahaman seperti itu, jadi diharapkan agar anda dapat memahami terlebih dahulu alasan dibalik pembuatan aplikasi ini sehingga mendapatkan kemudahan dalam mengikuti panduan pembuatan selanjutnya.

Untuk penjelasan lebih jelasnya dapat disimak pada pembahasan dibawah ini :

3.1.1 Latar Belakang Pembuatan Aplikasi

MSO (Management Service Operation) merupakan salah satu divisi yang terdapat pada PT. Telkom Indonesia yang memiliki peran dan tanggung jawab dalam pembangunan dan pemeliharaan jaringan Telkomsel dan lain-lain. Namun, pada pengimplementasiannya MSO masih memiliki kendala dimana mengarah kepada pelayanan yang ternyata terkadang kurang maksimal. Pelayanan kurang maksimal yang dimaksudkan adalah durasi penanganan gangguan jaringan pada suatu wilayah tidak konsisten. Dengan tidak kekonsistenannya divisi MSO pada saat penanganan suatu gangguan jaringan masih belum ada suatu sistem untuk melakukan proses klasterisasi penanganan gangguan tersebut.

Berdasarkan hal tersebut, dibuatlah sistem yang terkomputerisasi untuk pengelompokan penanganan gangguan jaringan baik dari segi wilayah, jenis gangguan, dan durasi penanganannya. K-means clustering akan dimanfaatkan untuk pengelompokan gangguan tersebut diselesaikan. Dalam hal tersebut akan menjadi sebuah perbaikan atau evaluasi pelayanan dari pihak PT. Telkom kepada customer khususnya Telkomsel.

3.1.2 Identifikasi Masalah Aplikasi

Berdasarkan dari latar belakang tersebut, rumusan masalah pada penelitian ini adalah sebagai berikut:

1. Bagaimana mengimplementasikan metode K-means clustering pada penelitian ini?
2. Apakah ada korelasi pada masing- masing cluster yang dihasilkan oleh sistem clustering?

3.1.3 Tujuan Aplikasi

Berdasarkan dari latar belakang tersebut, tujuan pada penelitian ini adalah sebagai berikut:

1. Mengimplementasikan metode K-means Clustering ke dalam suatu sistem Clustering Komplain Gangguan Jaringan Telkomsel
2. Menentukan korelasi pada masing- masing cluster yang dihasilkan oleh sistem clustering.

3.1.4 Manfaat Aplikasi

Penelitian ini diharapkan dapat memberikan manfaat yang berguna diantaranya sebagai berikut:

1. Mengetahui proses metode K-means dalam Sistem Clustering Komplain Gangguan Jaringan Telkomsel

2. Mengetahui klasterisasi gangguan jaringan dengan atribut yang digunakan yaitu jenis gangguan, wilayah, maupun durasi penanganannya

3.1.5 Ruang Lingkup Pembangunan Aplikasi

Untuk memudahkan perancangan aplikasi ini, tanpa mengurangi tujuan dari penelitian. Maka dijelaskan kedalam ruang lingkup sebagai berikut:

1. Metode yang digunakan adalah metode K-means Clustering
2. Penentuan K menggunakan metode Elbow menggunakan bahasa R
3. Hanya terbatas pada clusterisasi gangguan jaringan Telkomsel pada PT. Telkom Indonesia
4. Objek penelitian adalah customer Telkomsel

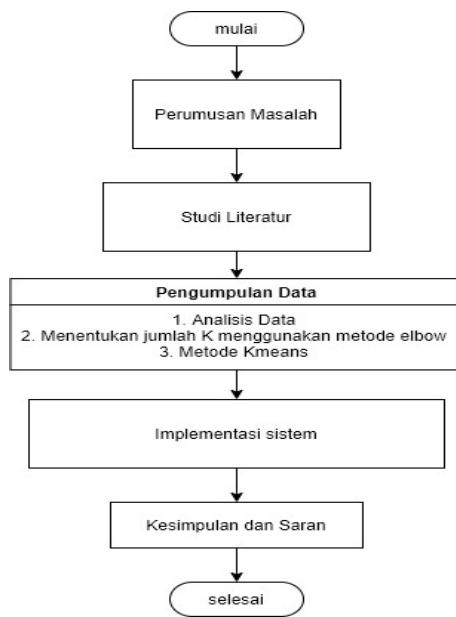
3.1.6 Kebutuhan Aplikasi

Pada setiap pembuatan aplikasi ada beberapa kebutuhan yang menjadi pedoman dalam membangun aplikasi itu sendiri. Untuk kebutuhan pembuatan aplikasi ini dapat dilihat dan dipahami sebagai berikut :

1. Kebutuhan analisis data yang sekiranya diperlukan dalam pembangunan aplikasi.

Kebutuhan analisis data sebelum penggerjaan aplikasi sangat penting dimana tujuannya unruk menghindari kesalahan sistematis sehingga kita tidak perlu mengulang program apabila ada kesalahan yang muncul akibat data ataupun pemahaman dasar. Untuk itu adapun kebutuhan analisis datanya mencakup hal berikut :

- Pengimplementasian konsep metodologi penelitian pengembangan aplikasi. Metodologi penelitian merupakan cara atau teknik yang digunakan dalam sebuah penelitian. Metodologi penelitian digunakan sebagai pedoman dalam pelaksanaan penelitian, agar hasil yang dicapai tidak menyimpang dari tujuan yang sebenarnya[3]. Penulis menggunakan metodologi penelitian agar dapat memudahkan proses pemecahan masalah penelitian.



Gambar 3.1 Diagram Alur Metodologi Penelitian

Adapun Penjelasan Diagram Alur Metodologi Penelitian :

(a) Perumusan Masalah

Perumusan masalah dilakukan untuk menentukan hal apa yang akan dibahas mengenai permasalahan-permasalahan yang ada pada PT. Telkom khususnya divisi *MSO (Management Service Operation)* yaitu pada proses penanganan gangguannya

(b) Studi Literatur

Selelah masalah penelitian dirumuskan, maka langkah kedua dalam proses penelitian ini adalah mencari teori-teori dan konsep-konsep hasil penelitian yang dapat dijadikan sebagai landasan teori untuk pelaksanaan penelitian. Studi literatur ini diperlukan dalam membantu proses penelitian sehingga peneliti dapat memperoleh berbagai sumber yang terkait dalam proses penelitian sehingga penelitian ini mempunyai teori yang kuat. Dalam hal ini, peneliti memperoleh informasi melalui buku, jurnal serta laporan penelitian untuk memperkuat teori dalam proses penelitian ini

(c) Pengumpulan Data

Observasi merupakan salah satu metode pengumpulan data yang melakukan pengamatan atau peninjauan langsung untuk mencari data-data yang dibutuhkan sesuai dengan permasalahannya yaitu membangun sistem sebagai monitoring dan penilaian kinerja pengembangan talent yang berbasis.

(d) Analisis Data

Data yang diambil untuk penelitian ini adalah data mentah gangguan jaringan . Untuk itu, agar data mentah yang diperoleh bisa berguna bagi penelitian ini, dan dilakukanlah analisis data. Analisa data yang dimaksudkan adalah memilih data apa saja yang berhak untuk diolah lebih lanjut nantinya ke dalam sistem.

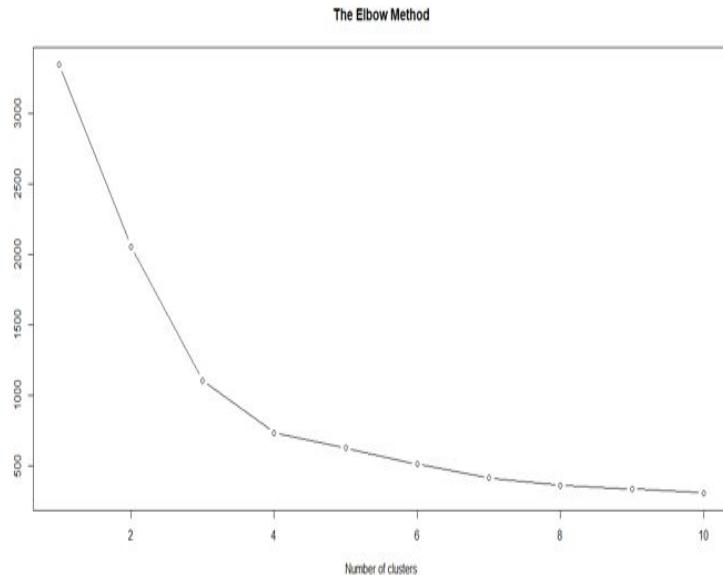
(e) Metode Elbow

Metode Elbow digunakan untuk penentuan K dalam metode clustering. Bahasa R digunakan untuk proses pengimplementasian Metode Elbow dengan menggunakan tools Rstudio.

Berikut dilampirkan source code dan hasil dari metode elbow.

```
1 dataset = read.csv('E:/SEMESTER 7/INTER1/LAPORAN/PRINT/500data.csv')
2 dataset = dataset[1:2]
3
4 # Menggunakan metode elbow untuk menemukan angka cluster yang optimal
5 set.seed(6)
6 wcss = vector()
7 for (i in 1:10){
8   wcss[i] = sum(kmeans(dataset, i)$withinss)
9 }
10 plot(1:10,
11       wcss,
12       type = 'b',
13       main = paste('The Elbow Method'),
14       xlab = 'Number of clusters',
15       ylab = 'wcss')
16
```

Gambar 3.2 Code Metode Elbow



Gambar 3.3 Hasil dari Metode Elbow

Jumlah K yang digunakan adalah 3 karena bentuk elbow (siku) terlihat saat jumlah kluster adalah 3($K=3$).

(f) Metode Kmeans Clustering

Data yang telah diperoleh maka akan diolah lebih lanjut ke dalam sistem clustering. Sistem tersebut dibuat guna mengimplementasikan metode Kmeans yaitu dalam hal pengelompokan gangguan jaringan

(g) Implementasi Sistem

Implementasi sistem tersebut sesuai dengan kebutuhan penelitian yang telah dilakukan, sehingga mempermudah peneliti untuk membangun sistem sesuai dengan apa yang dibutuhkan oleh perusahaan.

(h) Kesimpulan dan Saran

Hasil dari pengamatan dan penelitian yang telah dilakukan.

(a) Perangkat Perangkat yang Digunakan

Perangkat yang digunakan merupakan suatu cara agar dapat menghasilkan aplikasi yang sesuai dengan perangkat yang akan digunakan. Perangkat yang digunakan terdiri dari perangkat keras dan perangkat lunak

No	Nama Perangkat	Spesifikasi	Keterangan
1	<i>Memory</i>	4 GB	<i>Memory System</i> yang digunakan
2	<i>Processor</i>	<i>Intel® core™ i5-7200U CPU @ 3,18 Hz</i>	Untuk kecepatan transfer data dari sistem yang sangat bergantung pada kecepatan prosesor komputer
3	<i>System Type</i>	64-bit Operating System, x64-Based Processor	

Gambar 3.4 Perangkat Keras

(b) Perangkat Lunak Aplikasi yang digunakan

Adapun spesifikasi perangkat lunak (Software) yang digunakan untuk menjalankan aplikasi adalah sebagai berikut:

No	Tools / Software	Fungsi	Keterangan
1.	<i>Windows, Linux</i>	Sistem Operasi	-
2.	<i>MySQL</i>	Server Basis Data	-
3.	<i>Google Chrome, Mozilla Firefox, Internet Explorer</i>	Browser	-

Gambar 3.5 Perangkat Lunak

3.2 Pembangunan Awal Aplikasi

Pada pembangunan aplikasi ada beberapa hal yang harus dipersiapkan terlebih dahulu sehingga dalam pembangunannya akan lebih terinci dan tentunya lebih mudah untuk dikerjakan. Penjelasan lengkap akan dimuat pada pembahasan berikut sehingga aplikasi yang dibangun dapat berjalan dengan semestinya.

3.2.1 Instalasi Software Pembangun Aplikasi

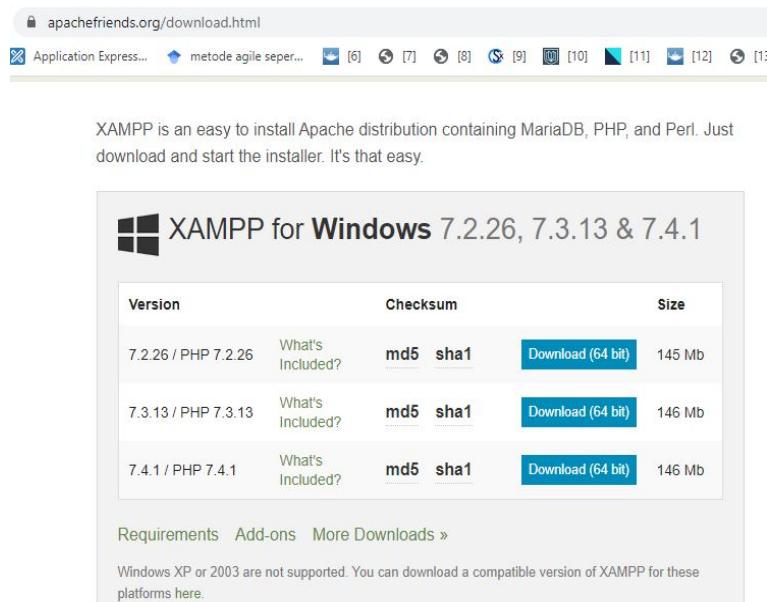
Pada pembahasan ini anda diharuskan untuk melakukan instalasi terhadap beberapa software yang menjadi pendukung dalam pembangunan aplikasi. Instalasi akan dijelaskan secara rinci dan mendetail sehingga anda dapat lebih mudah mengikuti panduan berikut. Software yang dibutuhkan ialah:

1. Instalasi XAMPP

Software pendukung yang pertamakali disediakan ialah XAMPP dimana merupakan distribusi apache kecil dan ringan yang berisi pengembangan web paling umum teknologi dalam satu paket. Isinya berupa ukuran kecil dan mudah dibawa menjadikannya alat yang ideal untuk pengguna mengembangkan dan menguji aplikasi dalam PHP dan MySQL. XAMPP tersedia secara gratis unduh dalam dua paket spesifik: lengkap dan ringan. Database driver yang kita gunakan ialah MySql. Mysql akan digunakan sebagai penyimpanan data selama aplikasi dibangun dan dijalankan baik itu data keperluan seperti master data, ataupun data tambahan lainnya[4].

Berikut langkah-langkah instalasi xampp :

- Download software XAMPP (mentahan)
- Anda dapat mendownload software tersebut dari link resmi xampp yaitu:
<https://www.apachefriends.org/index.html>.



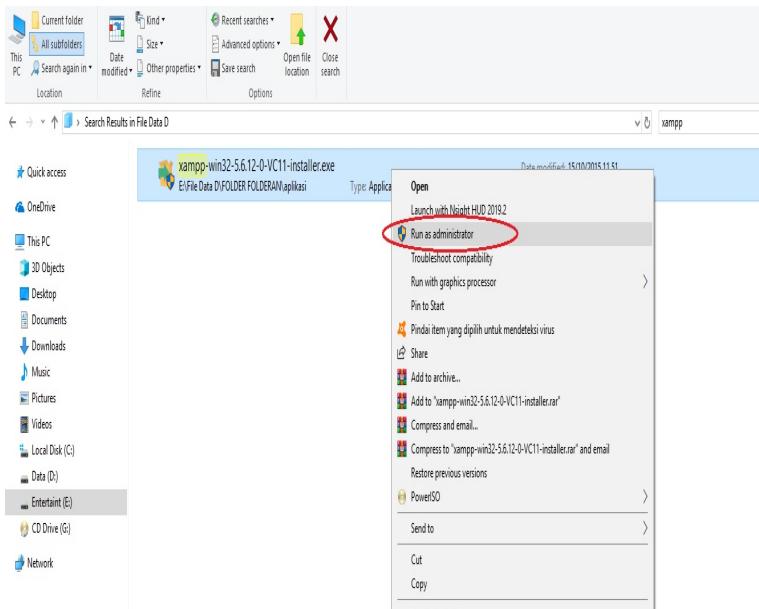
The screenshot shows the Apache Friends XAMPP download page for Windows. At the top, there's a navigation bar with links like Application Express..., metode agile seper..., [6], [7], [8], [9], [10], [11], [12], and [13]. Below the navigation bar, a message reads: "XAMPP is an easy to install Apache distribution containing MariaDB, PHP, and Perl. Just download and start the installer. It's that easy." The main content area features the XAMPP logo and the text "XAMPP for Windows 7.2.26, 7.3.13 & 7.4.1". Below this, there's a table with three rows of download links for different versions of XAMPP for Windows:

Version	Checksum	Size
7.2.26 / PHP 7.2.26	What's Included? md5 sha1	Download (64 bit) 145 Mb
7.3.13 / PHP 7.3.13	What's Included? md5 sha1	Download (64 bit) 146 Mb
7.4.1 / PHP 7.4.1	What's Included? md5 sha1	Download (64 bit) 146 Mb

At the bottom of the page, there are links for Requirements, Add-ons, and More Downloads ». A note at the very bottom states: "Windows XP or 2003 are not supported. You can download a compatible version of XAMPP for these platforms here."

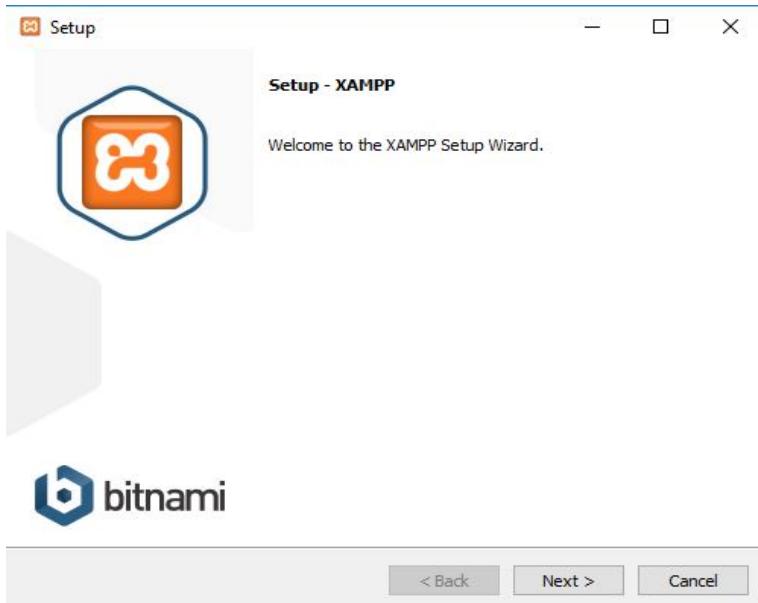
Gambar 3.6 Tutorial Instalasi XAMPP 01

- Pada pembuatan aplikasi ini, kita menggunakan XAMPP versi 3.2.1 silahkan proses download software disesuaikan.
- Setelah selesai mendownload software, maka silahkan lakukan instalasi.
- Instalasi dilakukan dengan cara menekan tombol kanan pada mouse / touchpad (klik kanan) lalu pilih run administrator seperti pada gambar dibawah:



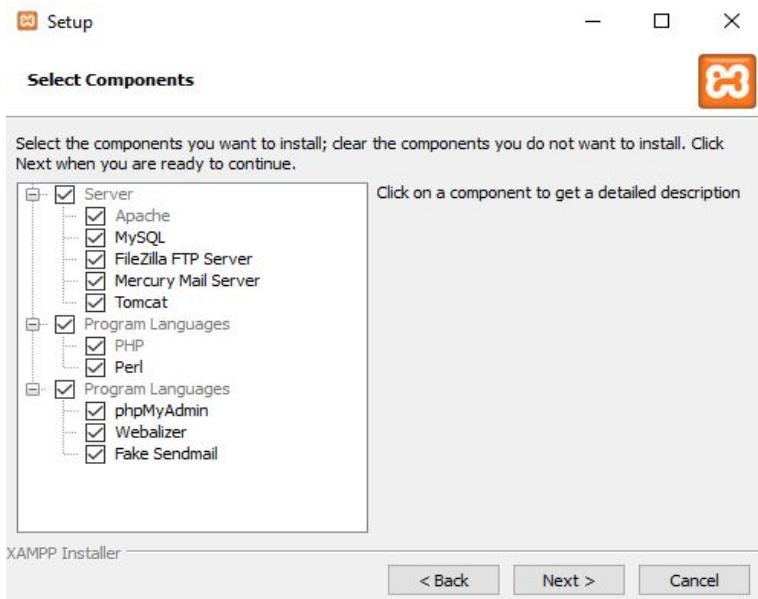
Gambar 3.7 Tutorial Instalasi XAMPP 02

- Setelah dijalankan, maka akan muncul tampilan dengan pilihan yes atau no. Silahkan klik button yes untuk melanjutkan instalasi.
- Selanjutnya untuk kelanjutan instalasi silahkan anda klik button next seperti pada tampilan instalasi berikut:



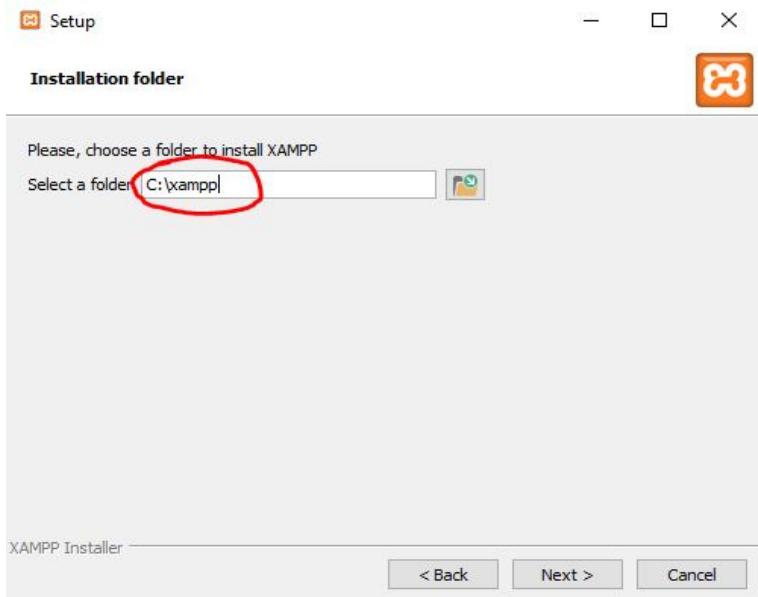
Gambar 3.8 Tutorial Instalasi XAMPP 03

- Silahkan klik button next kembali:



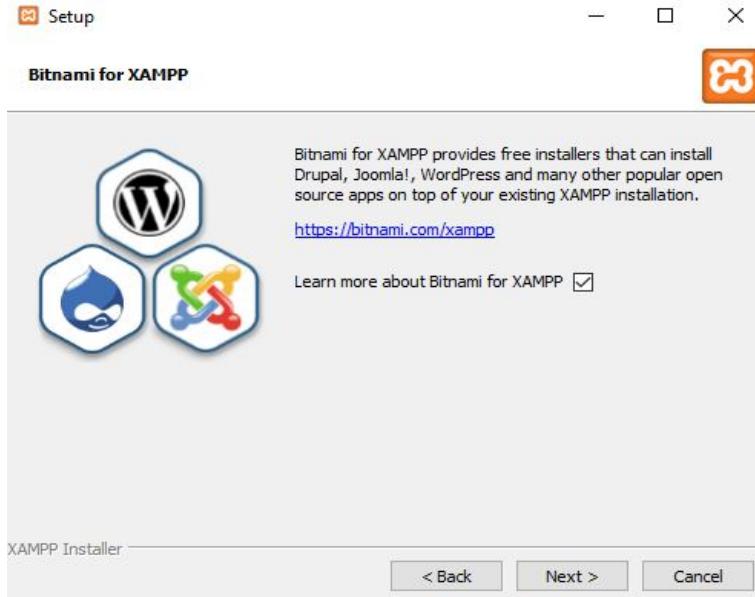
Gambar 3.9 Tutorial Instalasi XAMPP 04

- Selanjutnya pilih penyimpanan untuk software tersebut. Silahkan pilih penyimpanan pada drive C sehingga lebih efektif lalu klik button next. Untuk pemilihannya penyimpanannya dapat disesuaikan.



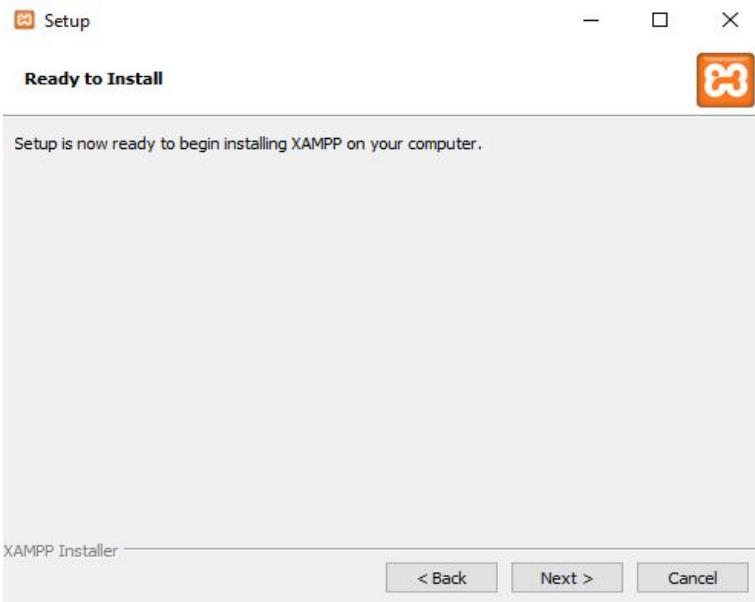
Gambar 3.10 Tutorial Instalasi XAMPP 05

- Klik button next untuk langkah selanjutnya.



Gambar 3.11 Tutorial Instalasi XAMPP 06

- Tampilan untuk kesiapan instalasi akan nampak seperti pada gambar dibawah dan silahkan klik kembali button next untuk melanjutkan proses.



Gambar 3.12 Tutorial Instalasi XAMPP 07

- Progress penginstalan akan tampak seperti gambar, anda hanya perlu menunggu beberapa saat sampai prosesnya selesai.



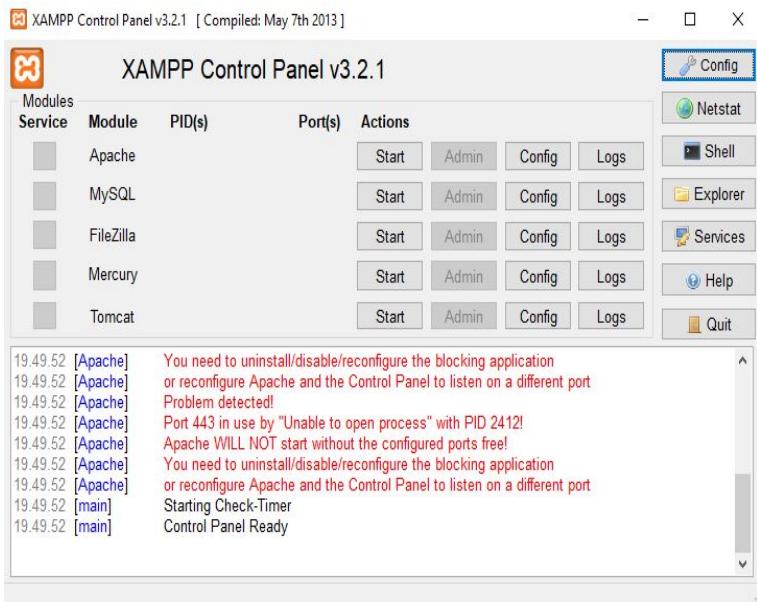
Gambar 3.13 Tutorial Instalasi XAMPP 08

- Setelah Progress penginstalan selesai, silahkan klik button finish untuk menyudahi proses secara keseluruhan.



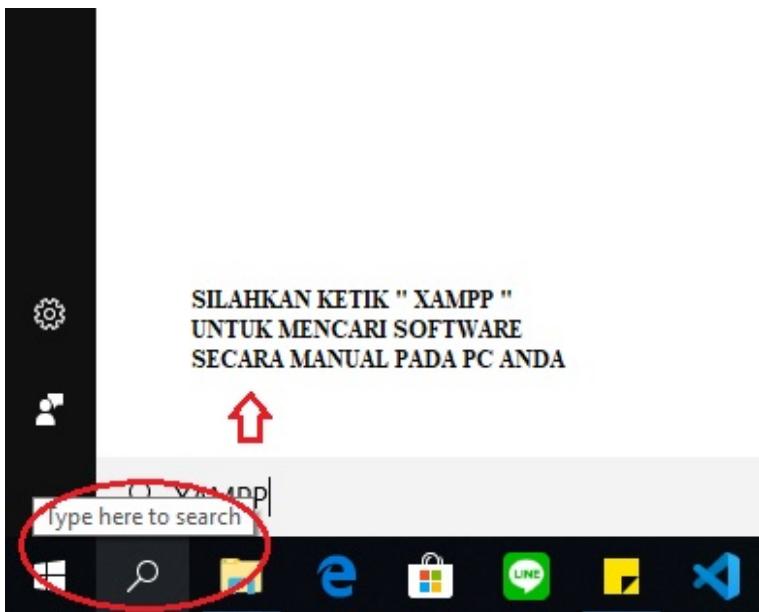
Gambar 3.14 Tutorial Instalasi XAMPP 09

- XAMPP akan otomatis terbuka apabila anda telah menyelesaikan proses instalasi, tampilannya akan nampak seperti pada gambar:



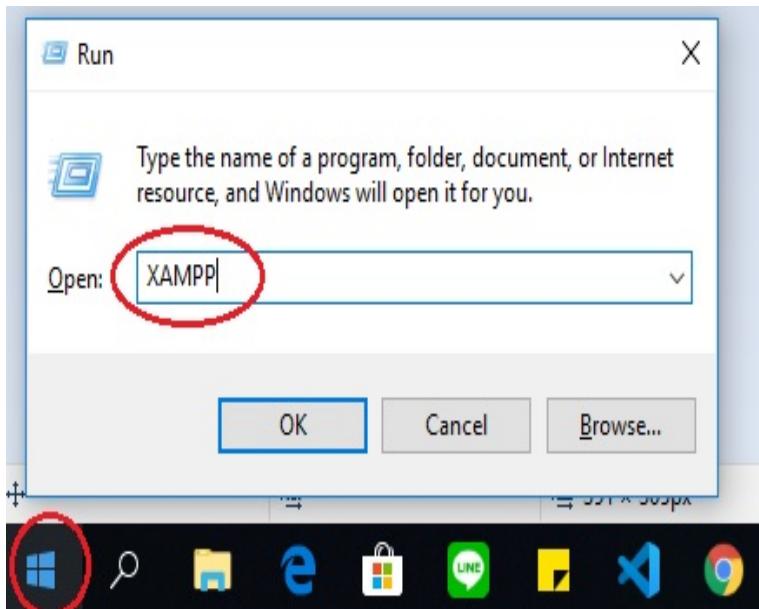
Gambar 3.15 Tutorial Instalasi XAMPP 10

- Apabila XAMPP tidak terbuka secara otomatis anda dapat membukanya secara manual yaitu dengan cara mencari software menggunakan dua cara yaitu:
 - (a) Mencari software melalui icon search pada toolbar laptop anda, contoh:



Gambar 3.16 Tutorial Instalasi XAMPP 11

- (b) Mencari melalui windows explorer yaitu menekan keyboard CTRL+R pada icon windows anda, contoh:



Gambar 3.17 Tutorial Instalasi XAMPP 12

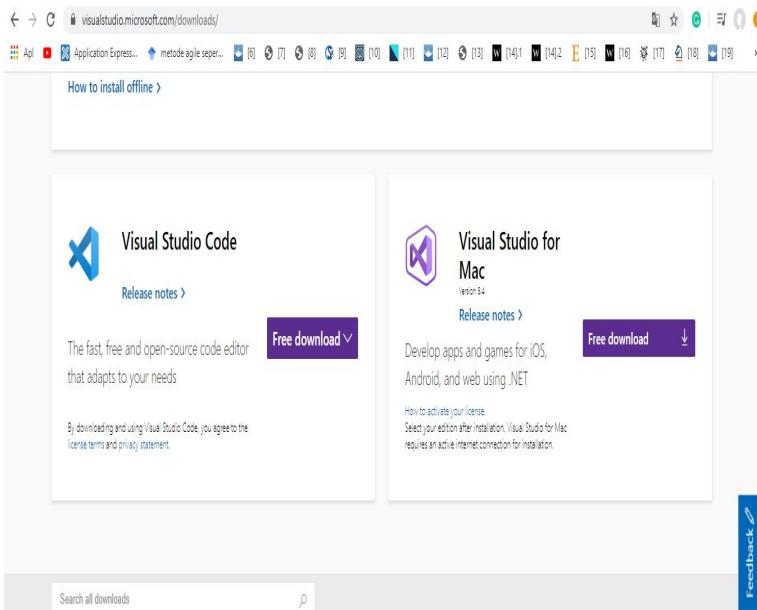
2. Instalasi Text Editor

Software pendukung selanjutnya yang harus disediakan ialah Text Editor. Text editor merupakan suatu software aplikasi atau suatu program komputer yang memungkinkan Anda sebagai penggunanya untuk membuat, mengubah atau mengedit file teks yang ada berupa plain text. Text editor ini sebenarnya bisa digunakan untuk membuat program-program komputer dan mengedit source code dari bahasa pemrograman.

Text Editor yang digunakan dalam panduan ini ialah Visual Studio Code. *Visual Studio Code* merupakan sebuah editor kode sumber yang dikembangkan oleh Microsoft untuk Windows, Linux, dan macOS. Fitur ini termasuk untuk debugging, kontrol Git yang tertanam dan GitHub dll. Software ini bersifat open source dan dirilis di bawah Lisensi MIT.

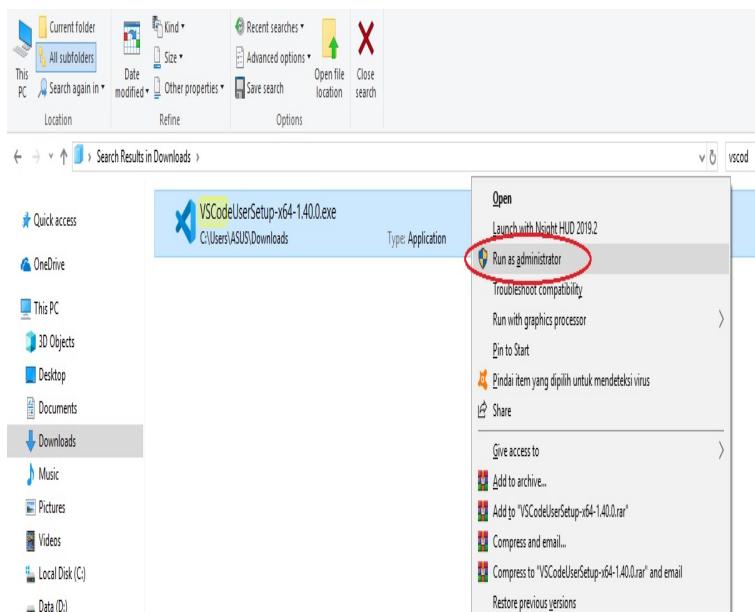
Berikut langkah-langkah instalasi Visual Studio Code :

- Download software Visual Studio Code (mentahan)
- Anda dapat mendownload software tersebut dari link resmi Visual Studio Code yaitu: <https://visualstudio.microsoft.com/downloads/>.



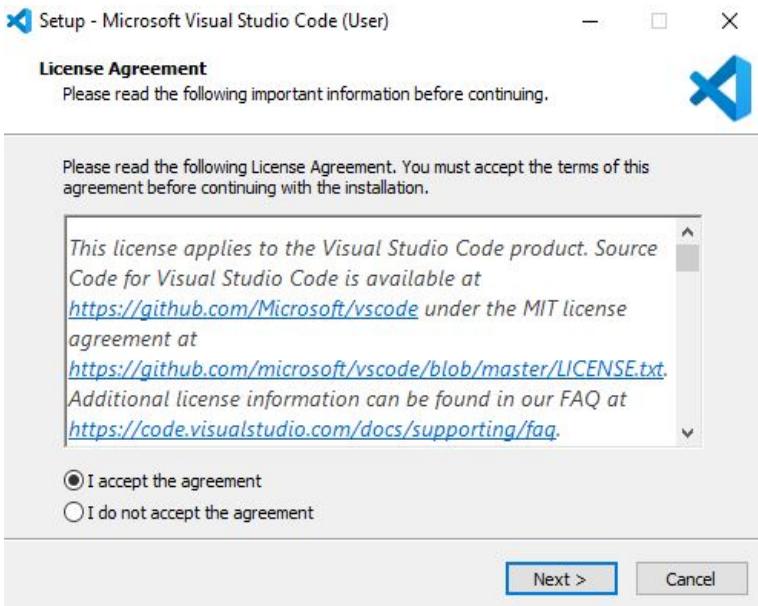
Gambar 3.18 Tutorial Instalasi Text Editor 01

- Pada pembuatan aplikasi ini, kita menggunakan Visual Studio Code versi terbaru. Silahkan proses download software disesuaikan.
- Setelah selesai mendownload software, maka silahkan lakukan instalasi.
- Instalasi dilakukan dengan cara menekan tombol kanan pada mouse / touchpad (klik kanan) lalu pilih run administrator seperti pada gambar dibawah:



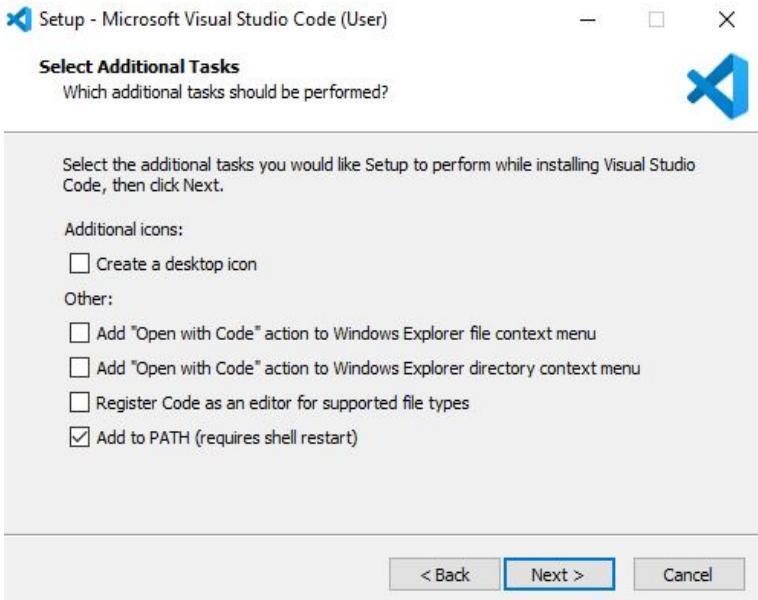
Gambar 3.19 Tutorial Instalasi Text Editor 02

- Setelah dijalankan, maka akan muncul tampilan dengan pilihan yes atau no. Silahkan klik button yes untuk melanjutkan instalasi.
- Selanjutnya untuk kelanjutan instalasi silahkan anda klik radio button *i accept the agreement* lalu klik button next seperti pada tampilan instalasi berikut:



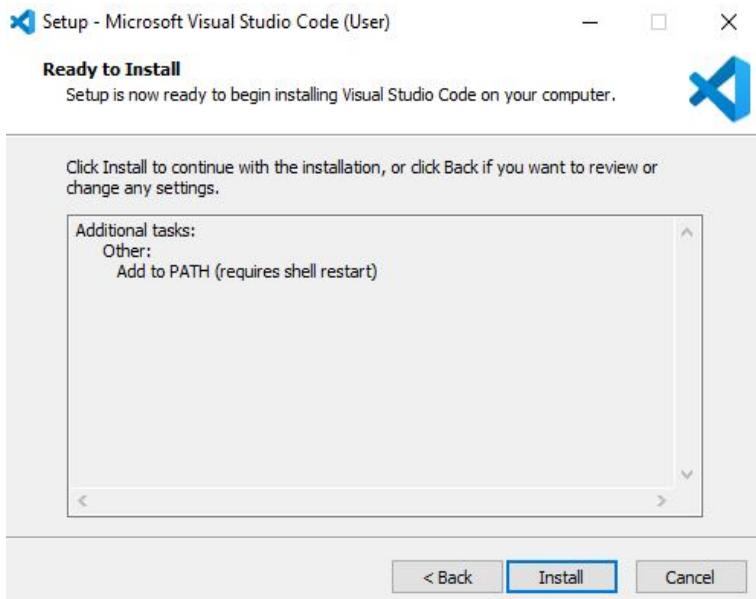
Gambar 3.20 Tutorial Instalasi Text Editor 03

- Selanjutnya, silahkan klik button next untuk proses berikutnya



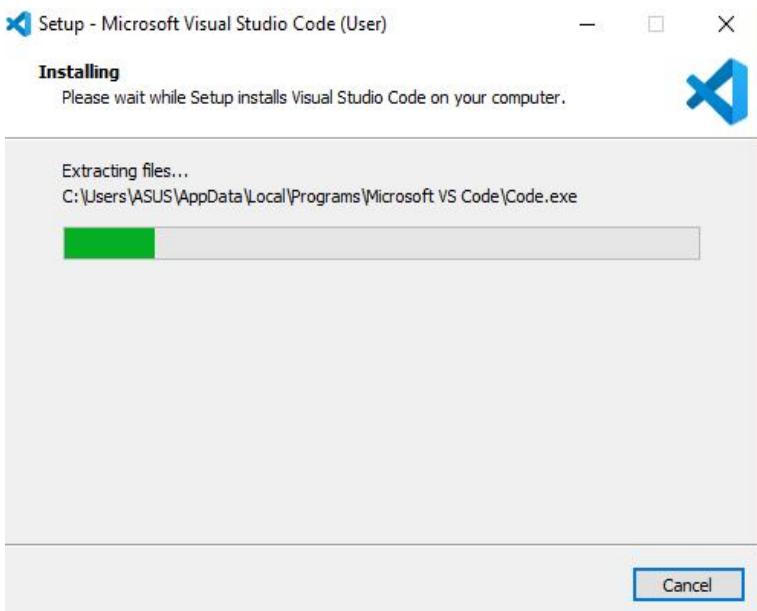
Gambar 3.21 Tutorial Instalasi Text Editor 04

- Silahkan klik button install untuk memulai proses instalasi text editor.



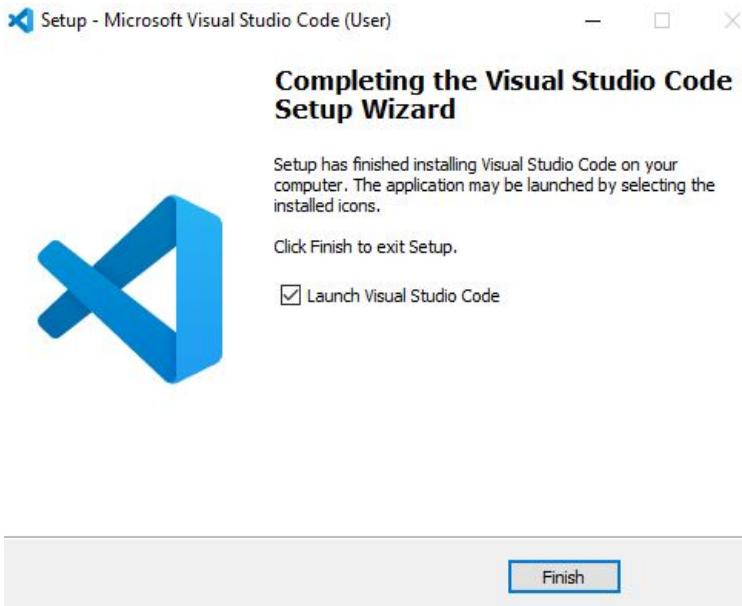
Gambar 3.22 Tutorial Instalasi Text Editor 05

- Progress pengistalan sedang berlangsung dan tampilannya akan nampak seperti pada gambar dibawah. Silahkan anda menunggu proses instalasi sampai selesai.



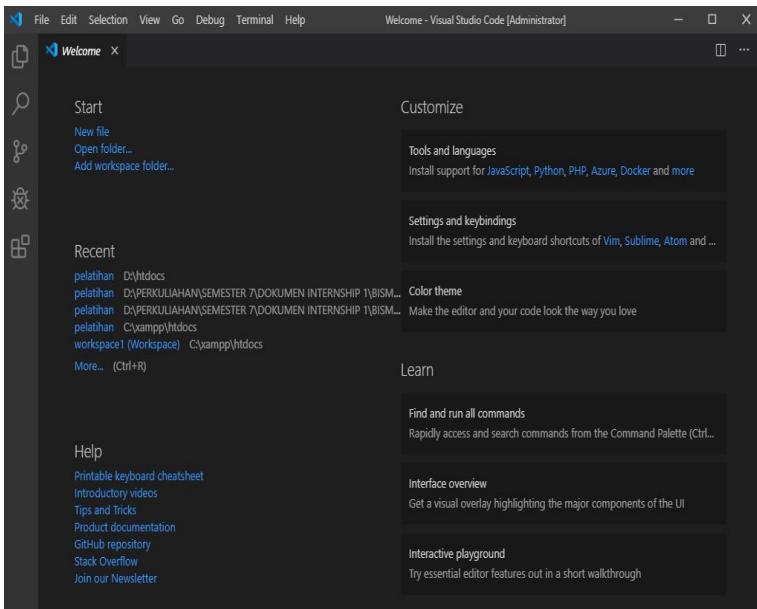
Gambar 3.23 Tutorial Instalasi Text Editor 06

- Ketika proses instalasi selesai dengan baik maka akan muncul tampilan seperti gambar. Silahkan anda menekan atau klik button finish untuk menyelesaikan instalasi secara keseluruhan.



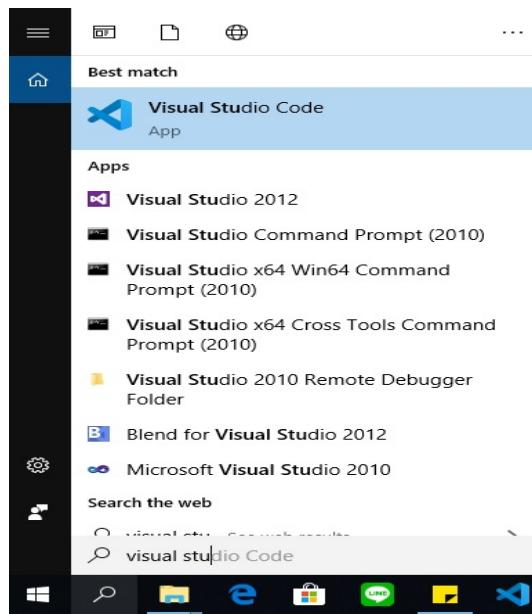
Gambar 3.24 Tutorial Instalasi Text Editor 07

- Visual Studio Code akan otomatis terbuka apabila anda telah menyelesaikan proses instalasi, tampilannya akan nampak seperti pada gambar:



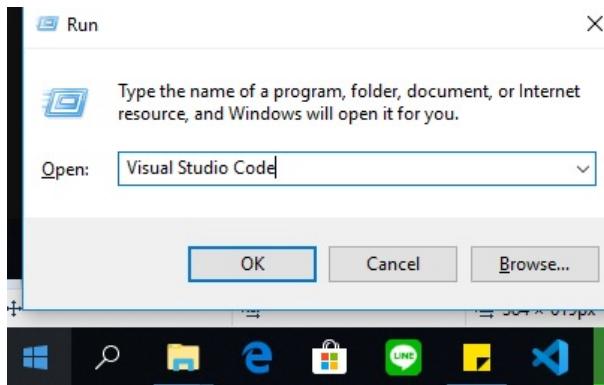
Gambar 3.25 Tutorial Instalasi Text Editor 08

- Apabila Visual Studio Code tidak terbuka secara otomatis anda dapat membukanya secara manual yaitu dengan cara mencari software menggunakan dua cara yaitu:
 - (a) Mencari software melalui icon search pada toolbar laptop anda, contoh:



Gambar 3.26 Tutorial Instalasi Text Editor 09

- (b) Mencari melalui windows explorer yaitu menekan keyboard CTRL+R pada icon windows anda, contoh:



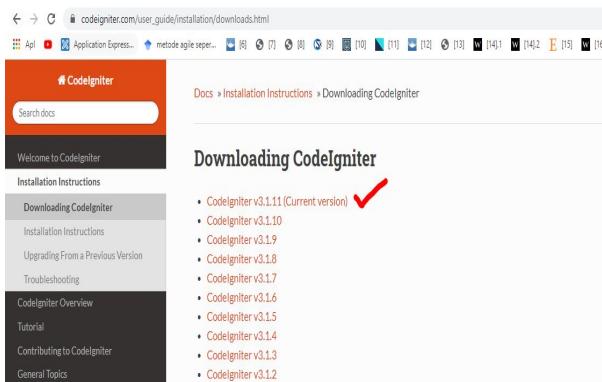
Gambar 3.27 Tutorial Instalasi Text Editor 10

3. Instalasi Codeigniter

Software berikutnya ialah Codeigniter. Codeigniter merupakan framework yang dimanfaatkan pada panduan pembangunan aplikasi ini. Penjelasan mengenai Codeigniter telah dibahas pada bab sebelumnya jadi kita bisa langsung memulai tahap-tahap penginstalan software.

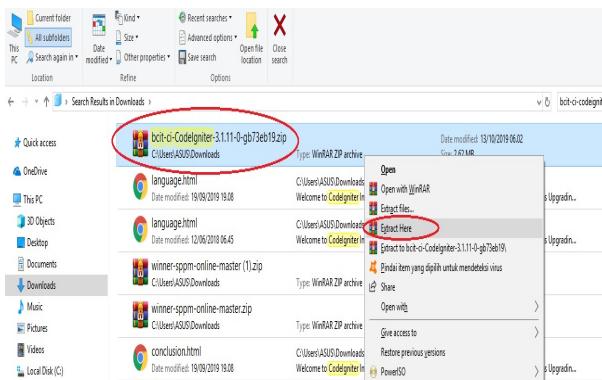
Berikut langkah-langkah instalasi Codeigniter :

- Download software Codeigniter (mentahan)
- Anda dapat mendownload software tersebut dari link resmi Codeigniter yaitu: <https://codeigniter.com/userguide/installation/downloads.html>.



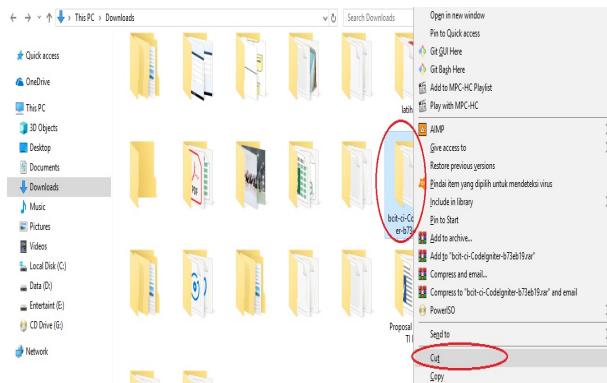
Gambar 3.28 Tutorial Instalasi Codeigniter 01

- Pada pembuatan aplikasi ini, kita menggunakan Codeigniter versi 3.1.11 (*the newest version*). Silahkan proses download software disesuaikan.
- Setelah selesai mendownload software, maka silahkan lakukan proses instalasi.
- Instalasi dilakukan dengan cara menekan tombol kanan pada mouse / touchpad (klik kanan) lalu pilih *extract here* untuk mengekstrak zip menjadi folder biasa sehingga filenya dapat digunakan. Lakukan seperti pada gambar berikut:



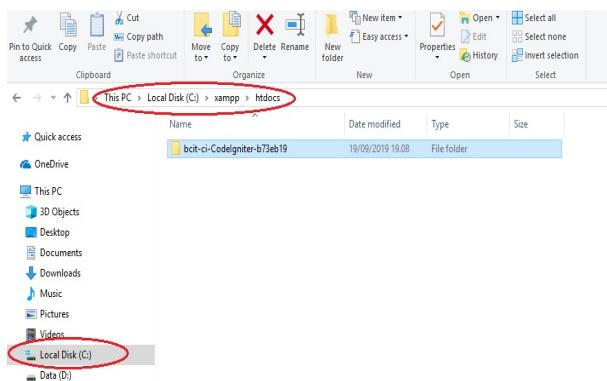
Gambar 3.29 Tutorial Instalasi Codeigniter 02

- Setelah melakukan ekstraksi data zip maka silahkan pindahkan folder kedalam folder htdocs yang ada dalam folder xampp pada drive C komputer anda :



Gambar 3.30 Tutorial Instalasi Codeigniter 03

- Tempatkan folder Codeigniter sesuai dengan gambar dibawah :



Gambar 3.31 Tutorial Instalasi Codeigniter 04

- Setelah penempatan folder, anda dapat melihat isi folder tersebut dimana berisikan seperti pada gambar berikut :

	Name	Date modified	Type	Size
Quick access	application	19/09/2019 19.08	File folder	
OneDrive	system	19/09/2019 19.08	File folder	
This PC	user_guide	19/09/2019 19.08	File folder	
3D Objects	editorconfig	19/09/2019 19.08	EDITORCONFIG File	1 KB
Desktop	gitignore	19/09/2019 19.08	Text Document	1 KB
Documents	composer.json	19/09/2019 19.08	JSON File	1 KB
Downloads	contributing.md	19/09/2019 19.08	MD File	7 KB
Music	index.php	19/09/2019 19.08	PHP File	11 KB
Pictures	license.txt	19/09/2019 19.08	Text Document	2 KB
Videos	readme.rst	19/09/2019 19.08	RST File	3 KB
Local Disk (C:)				
Data (D:)				

Gambar 3.32 Tutorial Instalasi Codeigniter 05

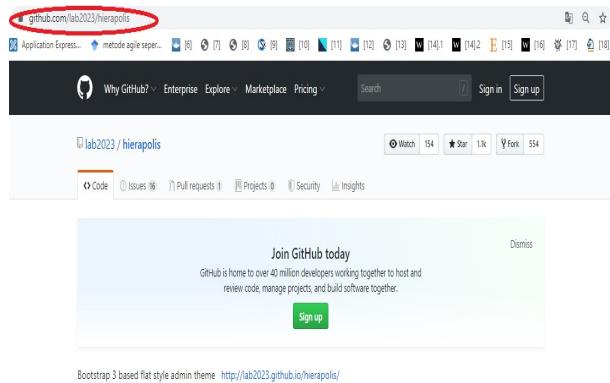
- Proses instalasi Codeigniter telah selesai dan mari beralih ke tahap selanjutnya.

4. Instalasi Boostrap (Template)

Software berikutnya ialah Bootstrap yaitu berupa template yang akan digunakan sebagai tampilan aplikasi berbasis web. Bootstrap merupakan kerangka CSS yang paling populer dalam mengembangkan situs web yang responsif dan bisa digunakan juga dalam mobile mode. Bootstrap merupakan produk open-source yang dibuat oleh Mark Otto dan Jacob T. yang ketika pertama kali dirilis, keduanya adalah karyawan di Twitter. Bootstrap diluncurkan pada Agustus 2011. Bootstrap telah berkembang sepenuhnya menjadi proyek yang digerakkan oleh CSS untuk memasukkan sejumlah plugin JavaScript dan ikon yang sejalan dengan bentuk dan tombol. Versi terbaru dari Boostrap ialah versi 4[5].

Berikut langkah-langkah instalasi Bootstrap :

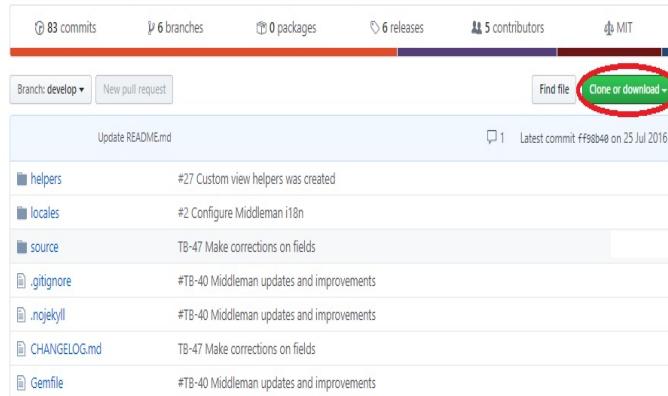
- Download software Booststrap (mentahan).
- Anda dapat mendownload software tersebut dari link atau website yang menyediakan *free-download*. Template yang akan didownload yaitu template Hierapolis yang termasuk Bootstrap Versi 3.
- Ada dua cara yang dapat anda gunakan untuk mendownload software, yaitu:
 - (a) Melalui akun github dengan link berikut : github.com/lab2023/hierapolis. Kunjungi akun github tersebut sesuai dengan contoh gambar dibawah:



Gambar 3.33 Tutorial Instalasi Bootstrap 01

- Pada akun github tersebut silahkan lakukan proses clone/download untuk menyimpan software Hierapolis pada PC anda.

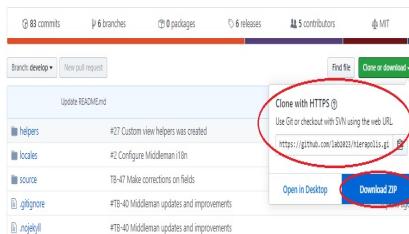
Bootstrap 3 based flat style admin theme <http://lab2023.github.io/herapolis/>



Gambar 3.34 Tutorial Instalasi Bootstrap 02

- Selanjutnya klik button download zip sesuai dengan gambar dibawah untuk melanjutkan proses download.

Bootstrap 3 based flat style admin theme: <http://lab2023.github.io/hierapolis/>

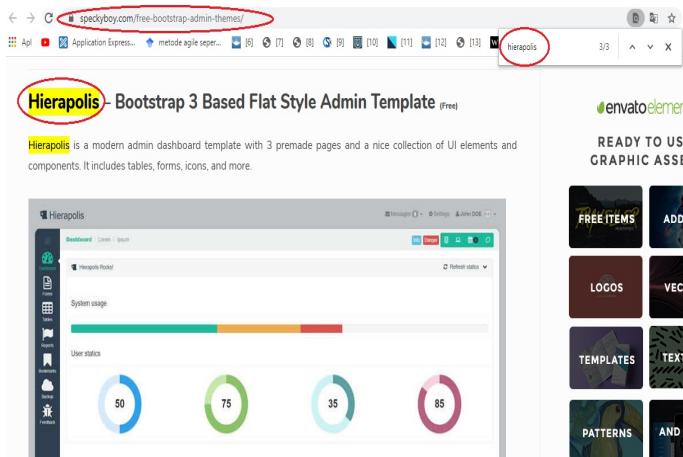


Gambar 3.35 Tutorial Instalasi Bootstrap 03

- Setelah melakukan seluruh proses tersebut maka software Hierapolis akan terdownload dan siap untuk dilakukan proses instalasi.
- (b) Melalui website dengan link berikut : speckyboy.com/free-bootstrap-admin-themes/. Kunjungi website tersebut sesuai dengan contoh gambar dibawah:

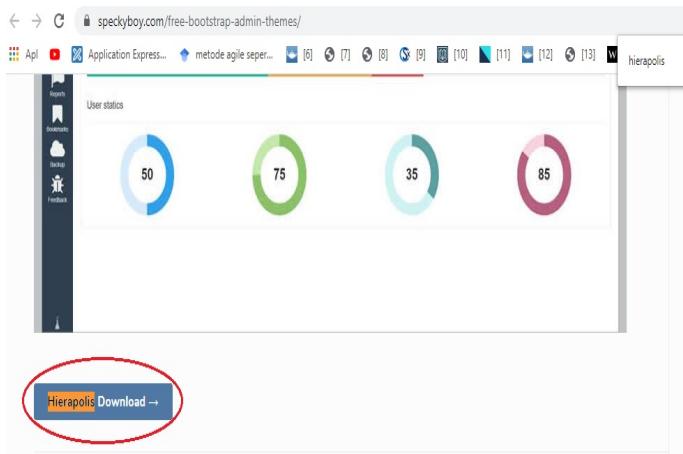
Gambar 3.36 Tutorial Instalasi Bootstrap 04

- Pada website tersebut terdapat beberapa template yang bisa digunakan, namun karena kita hanya menggunakan template Hierapolis maka kita akan mencari template tersebut.
- Silahkan *search* untuk template Hierapolis pada website dan tampilannya akan tampak seperti pada gambar :



Gambar 3.37 Tutorial Instalasi Bootstrap 05

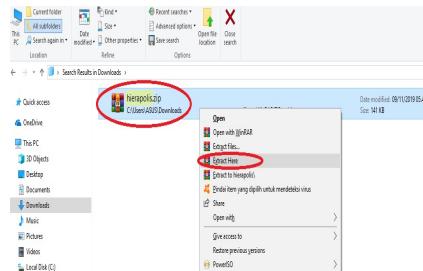
- Setelah template didapatkan silahkan klik button download pada bagian bawah template seperti contoh gambar :



Gambar 3.38 Tutorial Instalasi Bootstrap 06

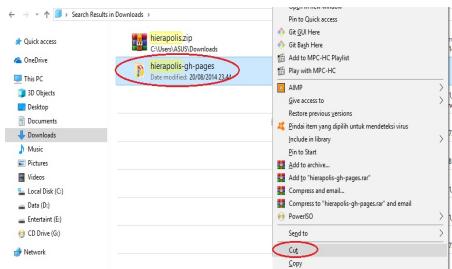
- Setelah melakukan seluruh proses tersebut maka software Hierapolis akan terdownload dan siap untuk dilakukan proses instalasi.
- Diantara kedua cara download diatas, penulis menyarankan untuk mengikuti proses kedua namun semua langkah-langkah yang dijelaskan dapat diikuti sesuai dengan kemudahan anda dalam menerapkannya.
- Proses Instalasi dilakukan dengan cara menekan tombol kanan pada mouse / touchpad (klik kanan) pada file Hierapolis yang telah didownload. Pilih

extract here untuk mengekstrak zip menjadi folder biasa sehingga filenya dapat digunakan. Lakukan seperti pada gambar berikut:



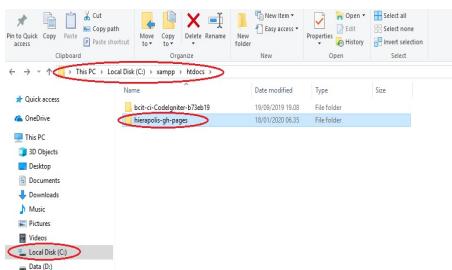
Gambar 3.39 Tutorial Instalasi Bootstrap 07

- Proses selanjutnya yaitu, memindahkan folder Hierapolis yang telah di ekstrak ke dalam folder htdocs yang berada dalam folder xampp pada drive C komputer anda.



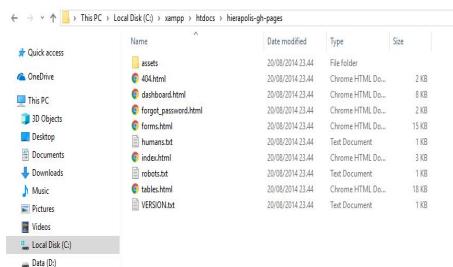
Gambar 3.40 Tutorial Instalasi Bootstrap 08

- Tempatkan folder Hierapolis sesuai dengan gambar dibawah :



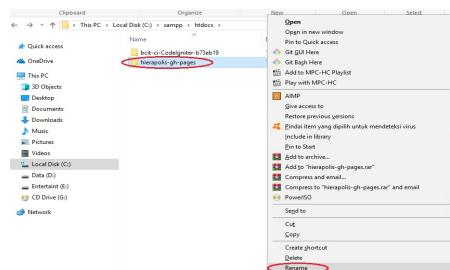
Gambar 3.41 Tutorial Instalasi Bootstrap 09

- Setelah penempatan folder, anda dapat melihat isi folder tersebut dimana berisikan seperti pada gambar berikut :



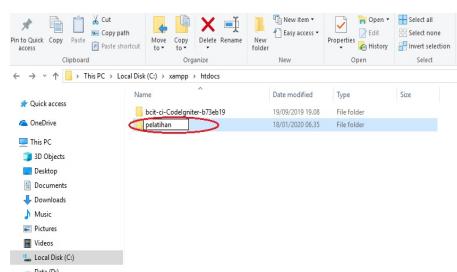
Gambar 3.42 Tutorial Instalasi Bootstrap 10

- Selanjutnya setelah semua proses dilakukan kita akan mengubah nama folder Hierapolis menjadi nama folder project yang dikerjakan. Nama project yang sedang dikerjakan ialah pelatihan jadi kita harus mengubah nama folder Hierapolis. Perubahan nama ini juga memudahkan kita dalam pembangunan aplikasi sehingga meminimalkan kesalahan dalam pemanggilan nama folder di panduan selanjutnya.
- Untuk perubahan nama dapat dilakukan dengan mengklik folder secara 2 kali atau dengan klik kanan pada folder kemudian pilih *rename* seperti pada gambar :



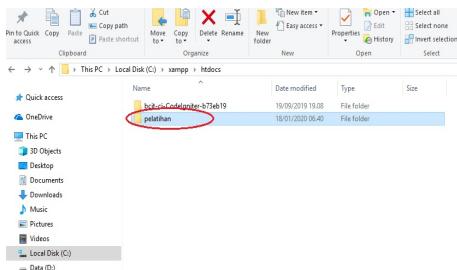
Gambar 3.43 Tutorial Instalasi Bootstrap 11

- Ubah nama folder dengan nama pelatihan sesuai gambar :



Gambar 3.44 Tutorial Instalasi Bootstrap 12

- Setelah perubahan tersebut maka tampilannya akan seperti ini :



Gambar 3.45 Tutorial Instalasi Bootstrap 13

- Proses instalasi Codeigniter telah selesai dan mari beralih ke tahap selanjutnya.

3.2.2 Panduan Penggunaan Awal Software Pendukung Aplikasi

Pada pembahasan ini anda akan dipandu dalam menjalankan software yang telah anda install sebelumnya. Panduan ini berguna untuk memberikan pemahaman tentang cara penggunaan software sehingga pada pembangunan aplikasi ini anda tidak merasa kesulitan dan tentunya menjadi pengalaman juga bagi anda dimana nantinya dapat diterapkan pada aktifitas lain sesuai kebutuhan. Untuk panduan ini akan dijelaskan secara lebih rinci dan mendetail dengan beberapa tahap. Silahkan simak tahapan berikut ini :

1. Penggunaan Awal XAMPP

Panduan pertama ialah penggunaan software XAMPP. XAMPP dijelaskan lebih awal karena software ini merupakan software yang sangat dibutuhkan dalam pembangunan aplikasi. Tanpa software ini, anda tidak dapat membuat *Database* dan tidak bisa menjalankan aplikasi (demo). Adapun pada panduan XAMPP berikut terdapat beberapa hal yang akan dijelaskan, silahkan simak penjelasan dibawah :

▪ Pengenalan Komponen XAMPP

Pertama-tama kita harus mengenal komponen pada XAMPP sehingga kita dapat menggunakan fitur yang ada pada XAMPP lebih baik. Terdapat beberapa komponen yang harus diperhatikan, yaitu :

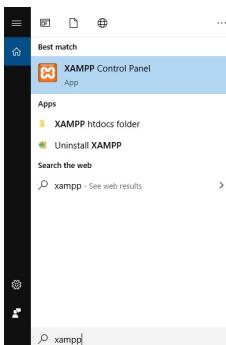
- (a) Apache : aplikasi web server default
- (b) MariaDB : sistem manajemen database
- (c) PHP : server side scripting untuk membuat aplikasi berbasis web
- (d) phpMyAdmin : tool untuk menggunakan MySQL berbasis web
- (e) OpenSSL : implementasi open-source dari dua protokol keamanan populer, yaitu SSL dan TSL

- (f) XAMPP Control Panel : kontrol panel sederhana untuk mengatur komponen berbeda pada XAMPP
- (g) Webalizer : sebuah tool analitik untuk user log dan metrik penggunaan
- (h) Mercury Mail Transport System : email server open source
- (i) FileZilla : berfungsi untuk melakukan transfer file
- (j) Tomcat : java servlet freeware untuk aplikasi Java
- (k) Strawberry Perl 7.0.56 Portable : berfungsi untuk melakukan distribusi Perl.

▪ Cara Menjalankan XAMPP

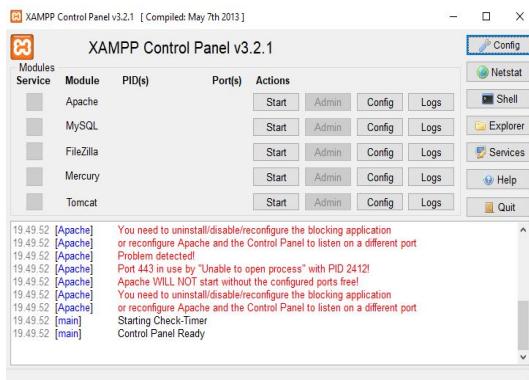
Berikut adalah cara menjalankan software XAMPP pada komputer anda:

- (a) Pertama-tama silahkan buka XAMPP yang telah di install pada pembahasan sebelumnya.
- (b) Cari software melalui windows explorer komputer anda selayaknya gambar dibawah :



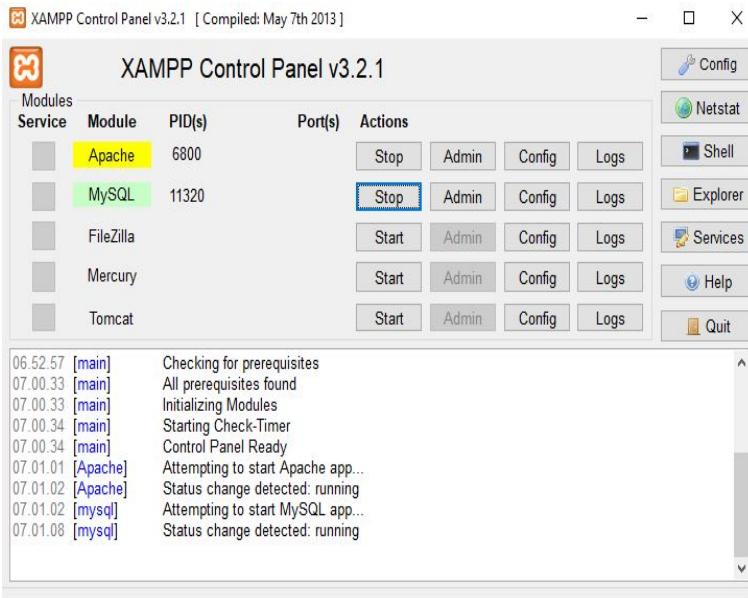
Gambar 3.46 Penggunaan XAMPP 01

- (c) Setelah aplikasi dibuka maka tampilannya akan seperti gambar dibawah:



Gambar 3.47 Penggunaan XAMPP 02

- Proses selanjutnya anda perlu menyalakan Apache dan MySQL pada XAMPP.
- Kedua fitur tersebut harus dinyalakan sehingga kita bisa melakukan pembangunan aplikasi dengan baik.
- Silahkan klik button start pada Apache dan MySQL seperti pada gambar dibawah :



Gambar 3.48 Penggunaan XAMPP 03

- (g) Setelah melaksanakan perintah tersebut maka XAMPP dapat digunakan:

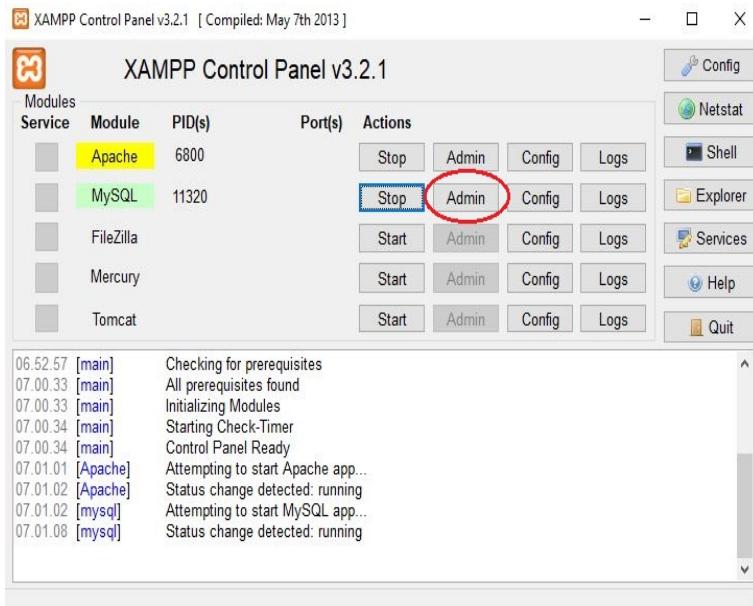
▪ Cara Menggunakan MySQL

Penggunaan MySQL sendiri untuk pembuatan *database*. *Database* adalah kumpulan informasi yang disimpan di dalam komputer secara sistematik sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Produk basis data seperti data open source MySQL yang digunakan oleh semakin banyak organisasi di dunia lebar.

Database MySQL adalah pilihan populer untuk pengembang basis data, administrator (DBAS), dan manajer TI yang menginginkan database kinerja tinggi yang andal, terjangkau, dan mudah menggunakan[6].

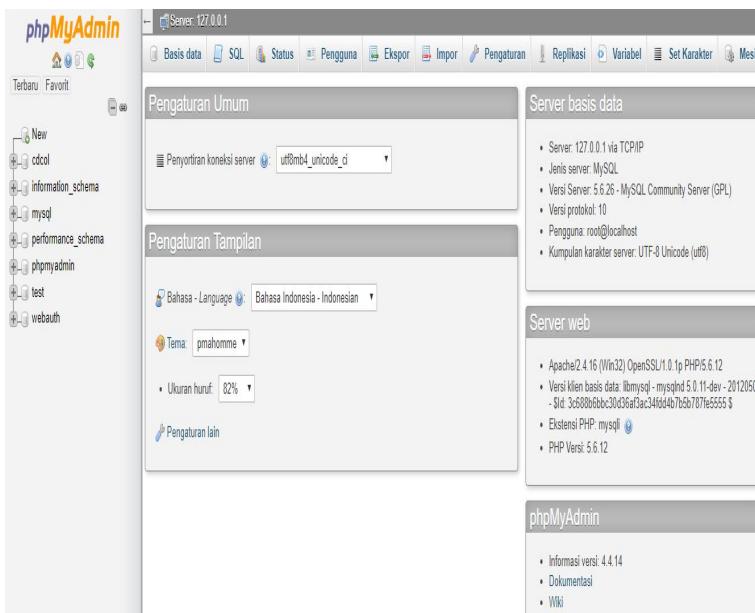
Untuk penjelasan lebih lengkap, simak langkah-langkah berikut :

- Pertama-tama silahkan buka kembali XAMPP
- Kemudian jalankan XAMPP
- Apabila tampilan sudah muncul, silahkan klik button seperti gambar dibawah untuk mengarahkan anda kepada phpMyAdmin dimana merupakan tempat untuk pembuatan MySql Database.



Gambar 3.49 Penggunaan XAMPP 04

- Berikut tampilan dari phpMyAdmin tempat anda dapat membuat *database* sesuai dengan kebutuhan aplikasi.



Gambar 3.50 Penggunaan XAMPP 05

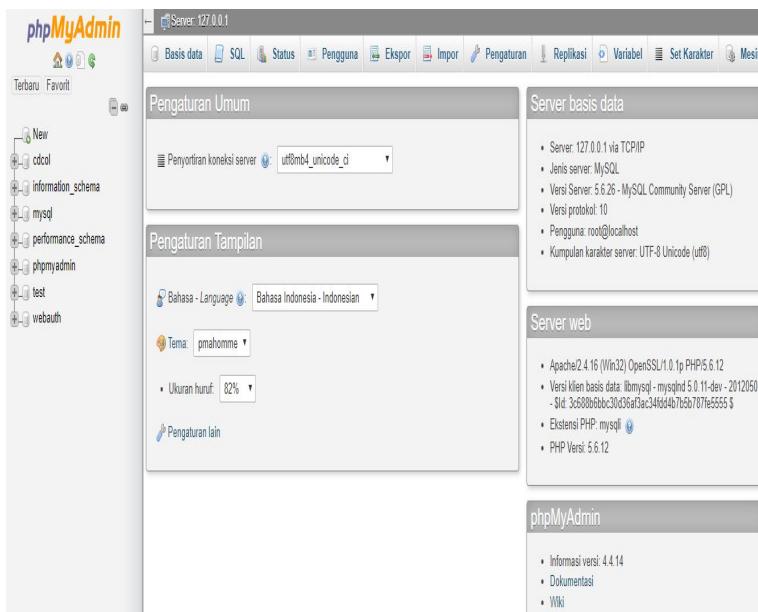
■ Panduan Penggunaan phpMyAdmin

Dalam penggunaan phpMyAdmin tentunya kita bisa membuat beberapa database dimana di dalamnya terdapat *tables* maupun *view*. Berikut penjelasan dan tata cara pembuatannya :

(a) Membuat Contoh Database

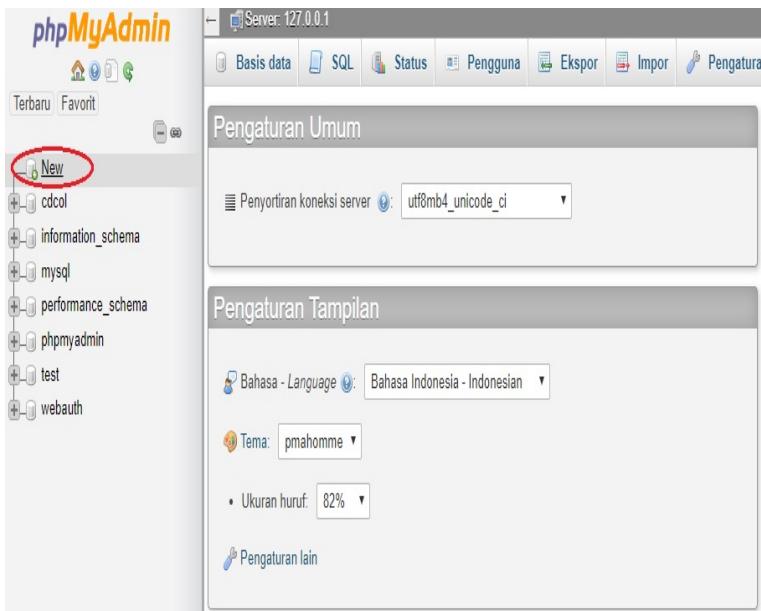
Dalam membuat *database* anda dapat mengikuti panduan berikut :

- Pertama-tama pastikan anda sudah berada pada phpMyAdmin seperti pada gambar :



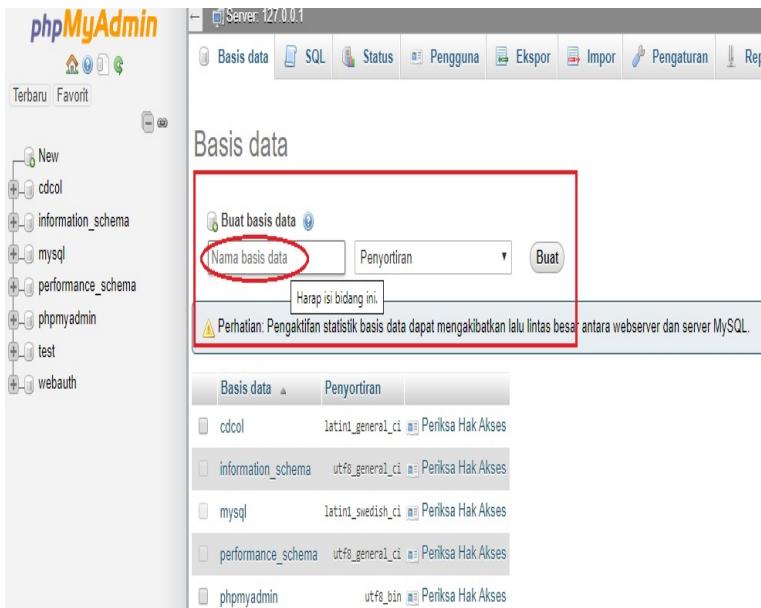
Gambar 3.51 Penggunaan XAMPP 06

- Selanjutnya silahkan klik button new untuk membuat *database* baru sesuai dengan kebutuhan anda ataupun aplikasi yang dibangun.



Gambar 3.52 Penggunaan XAMPP 07

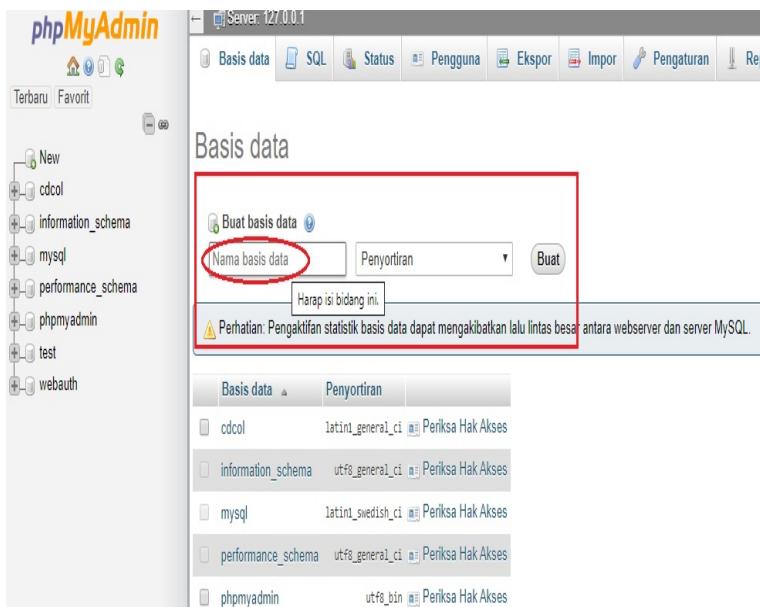
- Setelah menekan button *new* tampilannya akan seperti berikut:



Gambar 3.53 Penggunaan XAMPP 08

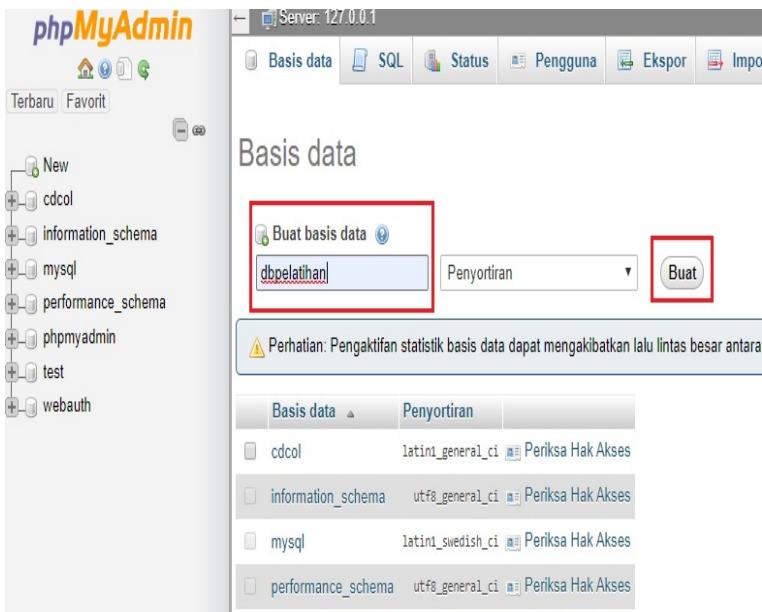
- Selanjutnya anda bisa mengisi kolom nama yang tertera.

- Silahkan anda isi nama dari yang akan dibuat. Pengisiannya terletak pada kolom berikut :



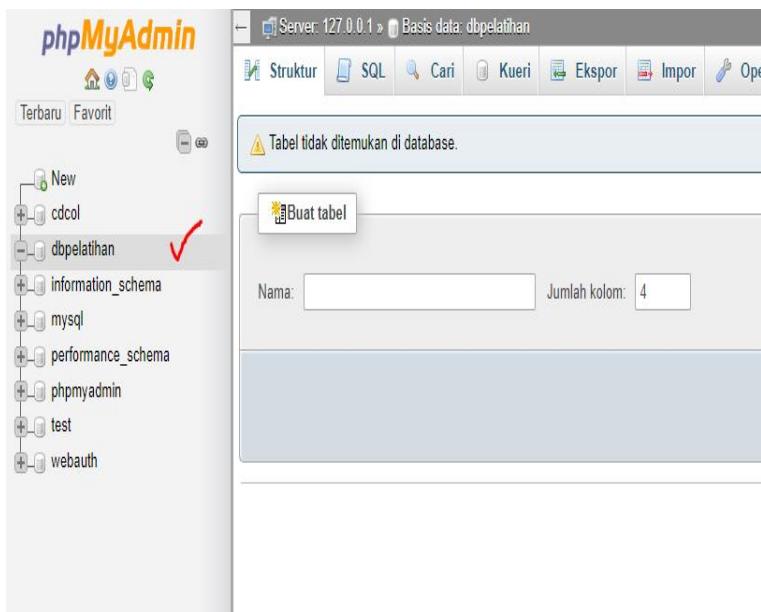
Gambar 3.54 Penggunaan XAMPP 09

- Pengisian nama *database* menggunakan nama dbpelatihan. Kita langsung saja sesuaikan nama database yang akan digunakan dalam pembangunan aplikasi sehingga kita tidak perlu membuat *database* kembali
- Pengisian nama dari telah dibuat maka silahkan klik button buat pada tampilan yang sama:



Gambar 3.55 Penggunaan XAMPP 10

- Selanjutnya setelah pembuatan *database* tersebut maka hasilnya akan nampak seperti gambar:

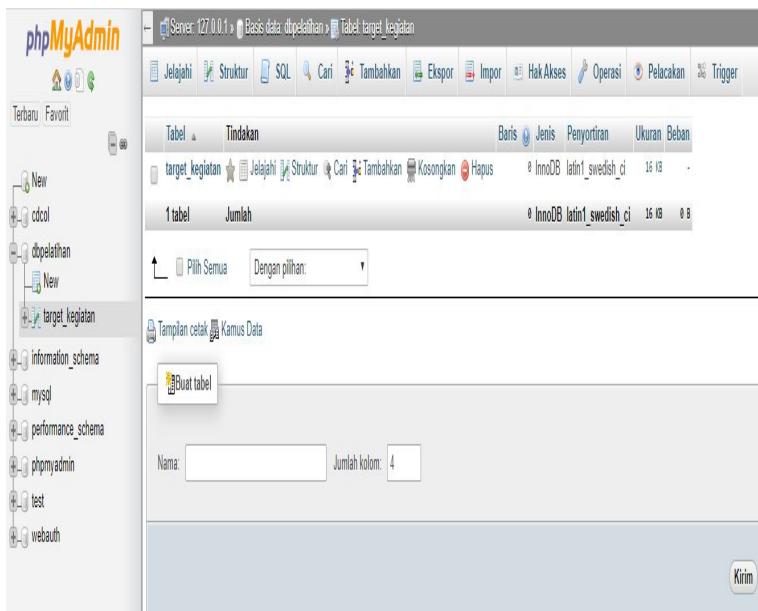


Gambar 3.56 Penggunaan XAMPP 11

(b) Membuat Contoh Table

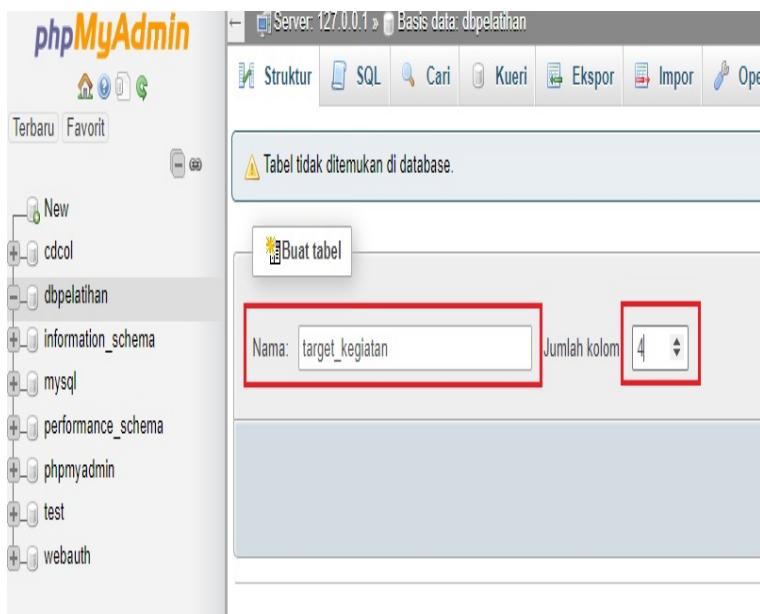
Dalam membuat *tables* anda dapat mengikuti panduan berikut :

- Pertama-tama pastikan anda berada dalam phpMyAdmin sehingga anda bisa melakukan proses selanjutnya.
- Selanjutnya anda klik button dbpelatihan yang telah dibuat sebelumnya



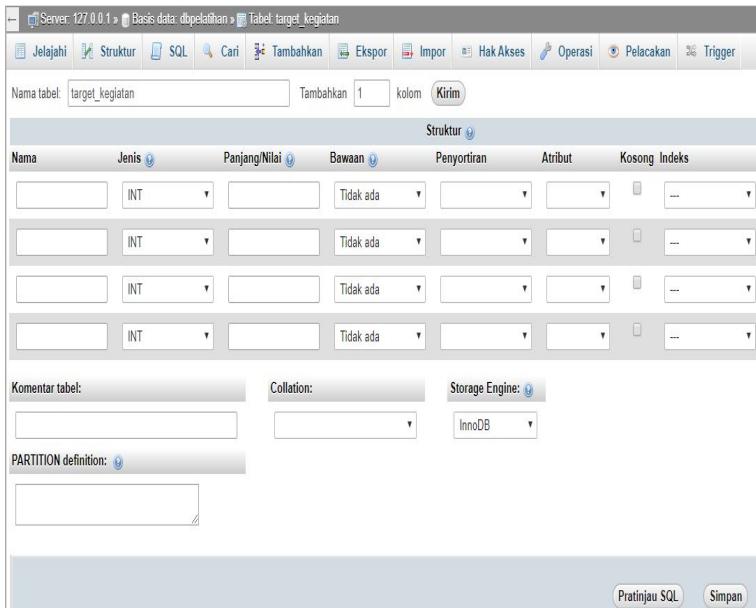
Gambar 3.57 Penggunaan XAMPP 12

- Silahkan isi kolom yang terdapat pada phpMyAdmin dan sesuaikan seperti gambar dibawah :

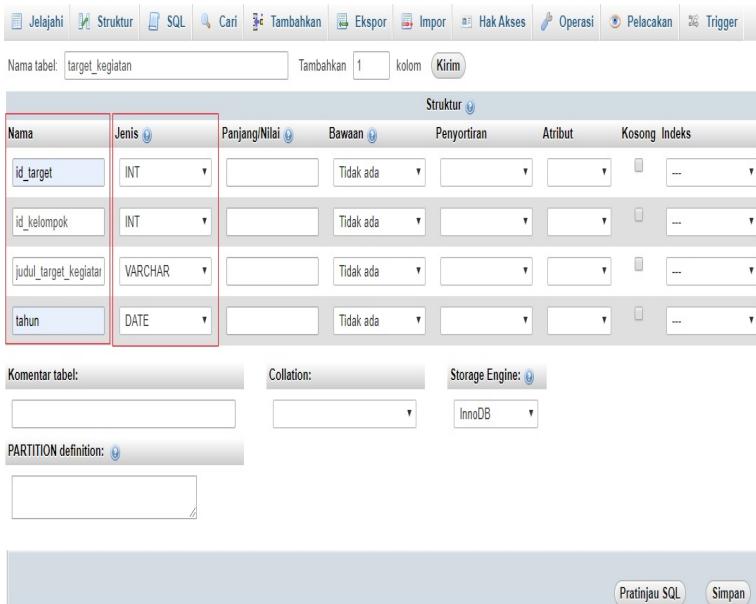


Gambar 3.58 Penggunaan XAMPP 13

- Pada gambar diatas dapat dilihat bahwa nama tabel yang digunakan ialah target kegiatan yang merupakan fitur yang ada dalam pembangunan aplikasi ini.
- Kita menggunakan nama target kegiatan agar nantinya kita tidak perlu membuat tabel kembali dan hanya tinggal menyesuaikan isi dari tabel tersebut.
- Untuk jumlah kolom dicontohkan menggunakan 4 kolom.
- Kolom ini dapat diganti dan disesuaikan pada pembangunan tabel *database* yang sebenarnya nanti, untuk saat ini dijadikan sebagai contoh saja agar anda memiliki pemahaman awal dalam pembuatan tabel *database*.
- Silahkan klik button simpan dibagian bawah pada halaman yang sama.
- Setelah disimpan maka tampilannya akan seperti berikut, yang mana kita diharuskan untuk mengisi field dari tabel tersebut.

**Gambar 3.59** Penggunaan XAMPP 14

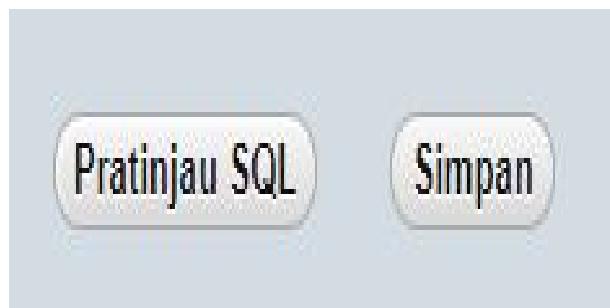
- Silahkan lakukan pengisian field sesuai dengan gambar dibawah:

**Gambar 3.60** Penggunaan XAMPP 15

- Pada pengisian perlu diperhatikan kolom nama yang merupakan field yang akan berhubungan dengan codingan pemrograman yang akan dipakai dalam pembangunan aplikasi.

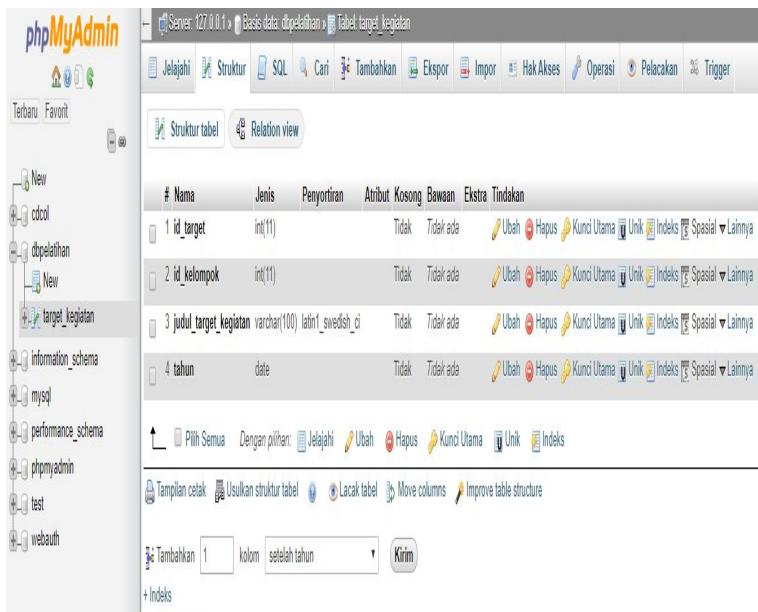
- Selain itu, anda juga harus memperhatikan jenis data dari setiap field yang dibuat apakah dia *varchar*, *date*, *char* dan lain sebagainya.

- Selanjutnya setelah semuanya dikerjakan silahkan klik button simpan untuk menyimpan field yang telah dibuat.



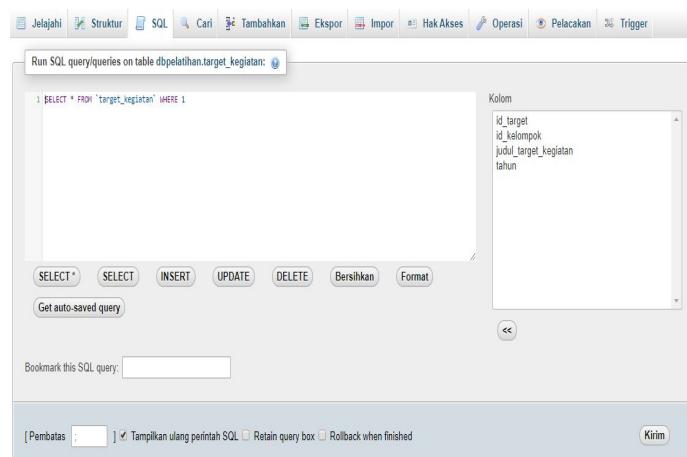
Gambar 3.61 Penggunaan XAMPP 16

- Tampilan dari field yang telah dibuat tadi akan melengkapi tabel seperti pada gambar berikut:



Gambar 3.62 Penggunaan XAMPP 17

- Adapun untuk pengisian dari field tersebut dapat dilakukan dengan beberapa cara.
- Yang pertama anda dapat menekan button SQL pada halaman yang sama dimana kita bisa mengisi query untuk eksekusi penambahan data. Setelah diklik maka tampilannya akan nam-pak seperti berikut:

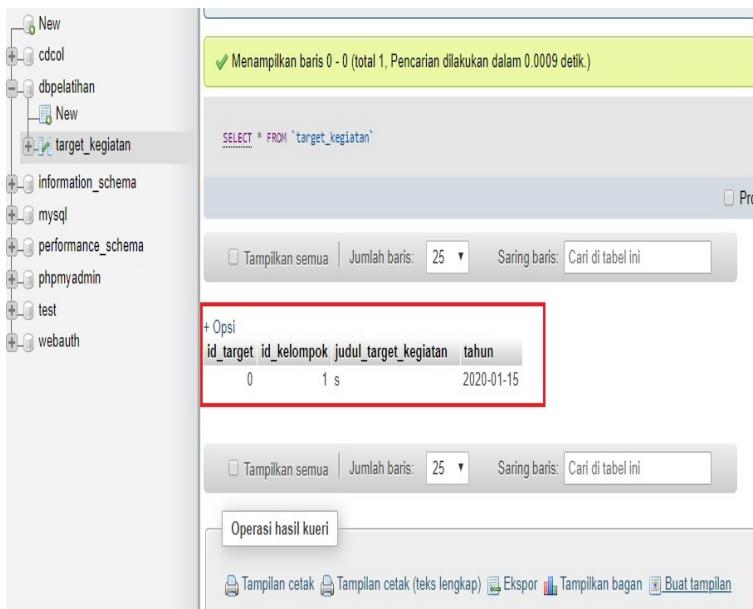


Gambar 3.63 Penggunaan XAMPP 18

- Cara yang lainnya yaitu dengan menekan button tambahkan pada halaman yang sama kemudian tampilannya akan nampak seperti berikut:

Gambar 3.64 Penggunaan XAMPP 19

- Setelah mengikuti tahap diatas, maka tampilannya akan nampak seperti gambar dibawah ini:



Gambar 3.65 Penggunaan XAMPP 20

- Tahapan selanjutnya ialah pemberian *primary key* untuk salah satu field yang menandakan bahwa field tersebut merupakan field yang unik dan menjadi kunci utama dalam tabel tersebut.
- Untuk *primary key* nantinya dapat digunakan untuk menghubungkan tabel dengan tabel lainnya apabila terdapat konsep join dan lain sebagainya sesuai dengan kebutuhan pembangunan aplikasi.
- Pembuatan *primary key* nampak seperti pada gambar yang mana anda harus menekan button *primary key* pada tabel terkait.

The screenshot shows the 'Struktur tabel' (Table Structure) tab in MySQL Workbench. The table 'target_kegiatan' is displayed with the following columns:

#	Nama	Jenis	Penyortiran	Atribut Kosong	Bawaan	Ekstra	Tindakan
1	id_target	int(11)		Tidak	Tidak ada	Ubah Hapus Kunci Utama	
2	id_kelompok	int(11)		Tidak	Tidak ada	Ubah Hapus Kunci Utama	
3	judul_target_kegiatan	varchar(100)	latin1_swedish_ci	Tidak	Tidak ada	Ubah Hapus Kunci Utama	
4	tahun	date		Tidak	Tidak ada	Ubah Hapus Kunci Utama	

Below the table, there are buttons for 'Pilih Semua' (Select All) and 'Dengan pilihan:' (With Selection), followed by icons for 'Jelajahi' (Explore), 'Ubah' (Edit), 'Hapus' (Delete), 'Kunci Utama' (Primary Key), 'Unik' (Unique), and 'Indeks' (Index).

Gambar 3.66 Penggunaan XAMPP 21

- Setelah menekan button tersebut maka akan muncul tampilan seperti ini dan anda hanya perlu menekan button oke untuk mengkonfirmasi eksekusi yang dilakukan.

The screenshot shows the 'Struktur tabel' (Table Structure) tab in MySQL Workbench. A confirmation dialog box is displayed, asking if the user wants to execute the following SQL command:

```
ALTER TABLE `target_kegiatan` ADD PRIMARY KEY(`id_target`);?
```

The dialog box has 'Oke' (OK) and 'Batal' (Cancel) buttons. The entire dialog box is highlighted with a red box.

Below the dialog box, the table structure is shown again with the 'id_target' column having a primary key icon.

Gambar 3.67 Penggunaan XAMPP 22

- Setelah konfirmasi maka tampilannya berubah dan telah terdapat *primary key* pada salah satu field dalam tabel tersebut.

The screenshot shows the MySQL Workbench interface. At the top, there is a green banner with the message: "MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 1.1671 detik.)". Below this, a SQL query is displayed: "ALTER TABLE `target_kegiatan` ADD PRIMARY KEY(`id_target`);". On the right, there are buttons for "Edit di kotak", "Ubah", and "Buat kode PHP". The main area shows a table structure for the "target_kegiatan" table:

#	Nama	Jenis	Penyortiran	Atribut Kosong	Bawaan	Ekstra	Tindakan
1	id_target	int(11)	Tidak	Tidak ada			Ubah Hapus Kunci Utama Unik Indeks Spasial Lainnya
2	id_kelompok	int(11)	Tidak	Tidak ada			Ubah Hapus Kunci Utama Unik Indeks Spasial Lainnya
3	judul_target_kegiatan	varchar(100) latin1_swedish_ci	Tidak	Tidak ada			Ubah Hapus Kunci Utama Unik Indeks Spasial Lainnya
4	tahun	date	Tidak	Tidak ada			Ubah Hapus Kunci Utama Unik Indeks Spasial Lainnya

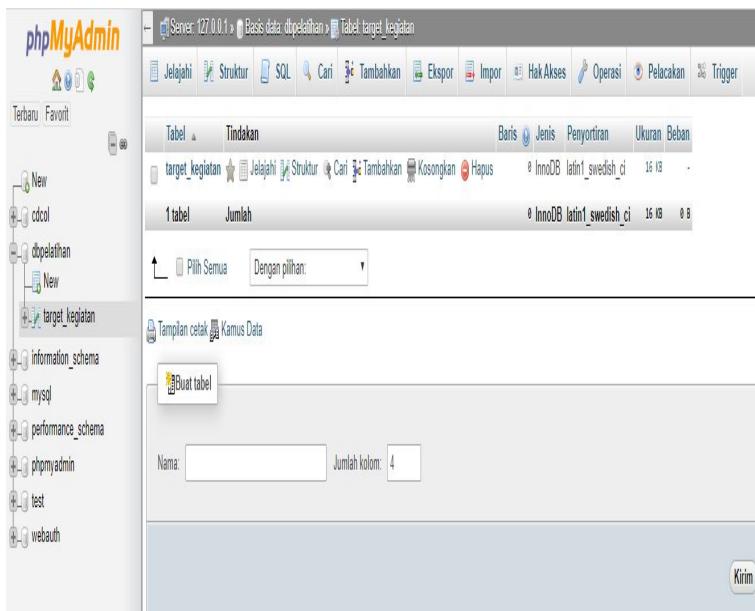
Below the table, there are navigation icons: up arrow, "Pilih Semua", "Dengan pilihan:", "Jelajahi", "Ubah", "Hapus", "Kunci Utama", "Unik", and "Indeks".

Gambar 3.68 Penggunaan XAMPP 23

(c) Membuat Contoh View

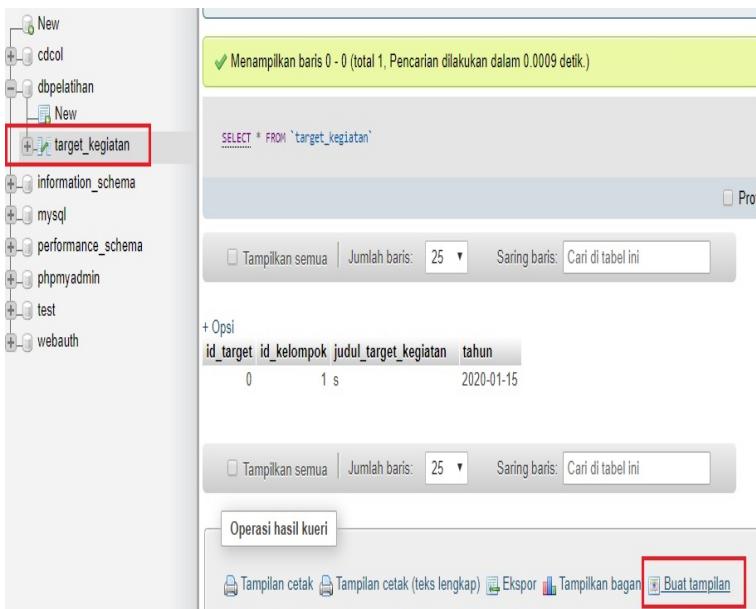
Dalam membuat *view* anda dapat mengikuti panduan berikut :

- Pertama-tama silahkan buka tabel target kegiatan yang telah dibuat sebelumnya



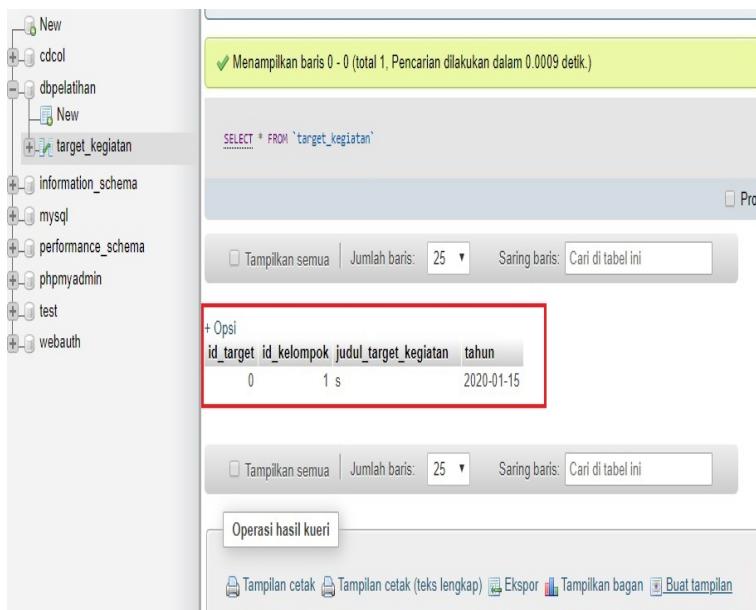
Gambar 3.69 Penggunaan XAMPP 24

- Selanjutnya anda dapat memilih button membuat tampilan pada bagian bawah halaman yang sama. Untuk lebih jelasnya silahkan anda perhatikan gambar berikut :



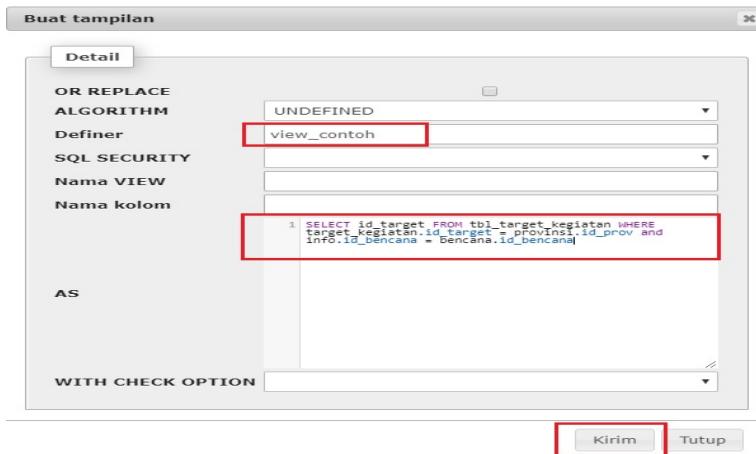
Gambar 3.70 Penggunaan XAMPP 25

- Hal yang harus diperhatikan ialah, button tersebut akan muncul apabila dalam tabel yang dipilih memiliki data sehingga dapat dilakukan eksekusi. Maka dari itu, kita harus menyediakan data untuk contoh seperti pada gambar dibawah:



Gambar 3.71 Penggunaan XAMPP 26

- Setelah anda menekan button membuat tampilan maka halaman yang akan muncul nampak seperti ini:

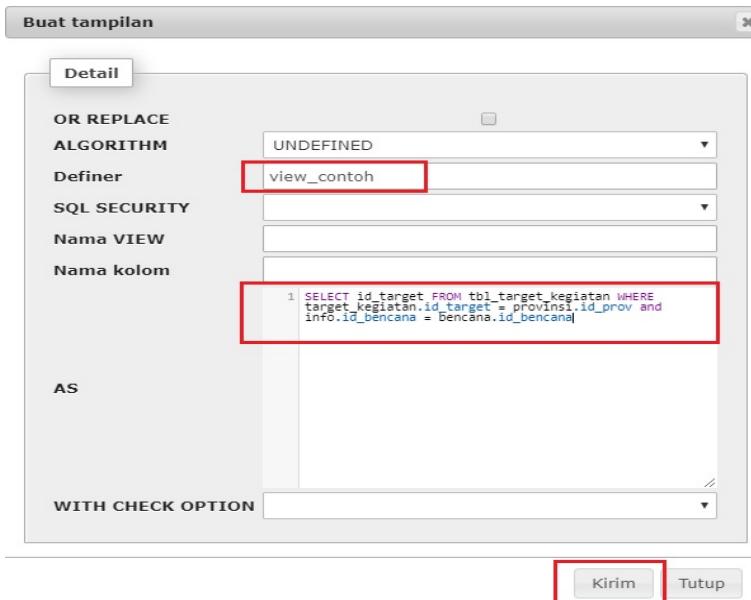


Gambar 3.72 Penggunaan XAMPP 27

- Yang harus diperhatikan ialah kolom definer yang harus diisi
- Kolom As juga penting untuk diisi dimana di dalamnya akan dimasukkan query yang menerapkan konsep join pada tabel

sehingga membuat tampilan sedemikian rupa berdasarkan tabel yang ada.

- Queri contoh yang tertera pada gambar diatas tidak bisa dieksekusi dikarenakan kita hanya memiliki satu tabel apabila ingin menjalankannya silahkan membuat tabel baru dan anda dapat mencoba untuk menggabungkan kedua tabel yang telah dibuat.
- Untuk penyelesaian eksekusi silahkan anda klik button kirim yang ada pada halaman yang sama seperti pada gambar :



Gambar 3.73 Penggunaan XAMPP 28

- (d) Setelah perintah dijalankan maka seluruh proses pembuatan *view* telah selesai.

▪ Manfaat XAMPP

Pada penggunaan XAMPP, ada manfaat yang bisa didapatkan sebagai pendukung pembangunan aplikasi seperti berikut:

- (a) XAMPP tersedia dalam GNU *General Public License* dan bebas dimana merupakan sebuah web server yang mudah digunakan yang menampilkan halaman web yang dinamis. Karena memiliki keterselektifan yang baik maka tentu saja XAMPP sangat efektif dan efisien dalam penggunaanya.
- (b) XAMPP memuat server HTTP Apache atau server web/www apache yang mana merupakan server web yang dapat dijalankan di

banyak sistem operasi menyerupai (Unix, BSD, Linux, Microsoft Windows dan Novell Netware serta platform lainnya) yang memiliki kegunaan untuk melayani dan memfungsikan situs web. Protokol yang digunakan untuk melayani kemudahan web/www ini menggunakan HTTP. XAMPP dinilai sangat bermanfaat bagi pembuatan program yang lebih luas dan juga dinamis.

2. Penggunaan Awal Text Editor

Panduan selanjutnya ialah penggunaan software Text Editor. Text Editor yang digunakan ialah *Visual Studio Code*. Seperti yang telah dijelaskan sebelumnya pada panduan penginstalan text editor, penggunaan *Visual Studio Code* ini akan sangat membantu dalam pembangunan aplikasi yang dikerjakan. Adapun pada panduan *Visual Studio Code* berikut terdapat beberapa hal yang akan dijelaskan, silahkan simak penjelasan dibawah :

▪ Pengenalan Plugins *Visual Studio Code*

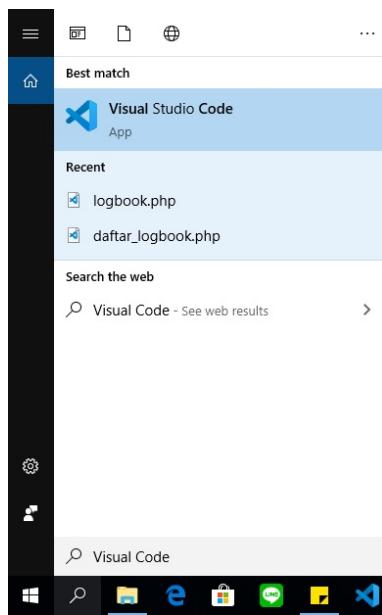
Pada pembahasan ini, kita akan belajar mengenai plugins apa saja yang dapat digunakan pada text editor VCS yang pastinya akan membantu, mempermudah, mempercantik program yang sedang dibangun. Plugin sendiri merupakan sebuah script tambahan yang dirancang untuk melakukan fungsi tertentu pada data yang sedang anda kerjakan.

▪ Cara Menjalankan *Visual Studio Code*

Pada pmbahasan ini anda akan belajar cara menggunakan *Visual Studio Code* sehingga pada saat penggerjaan aplikasi anda telah lebih paham untuk menggunakannya.

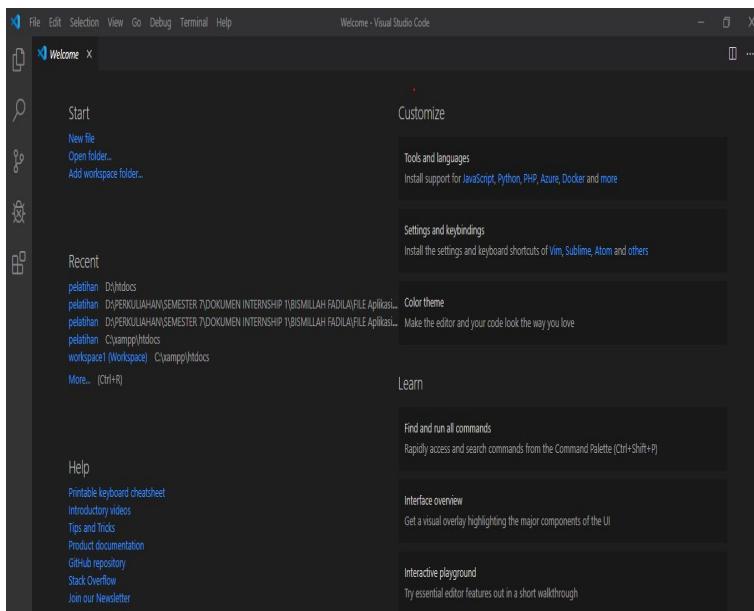
Silahkan simak beberapa penjelasan dibawah ini:

- (a) Pertama-tama silahkan buka software *Visual Code Studio* pada komputer anda
- (b) Anda dapat membuka software VSC dengan cara seperti ini:



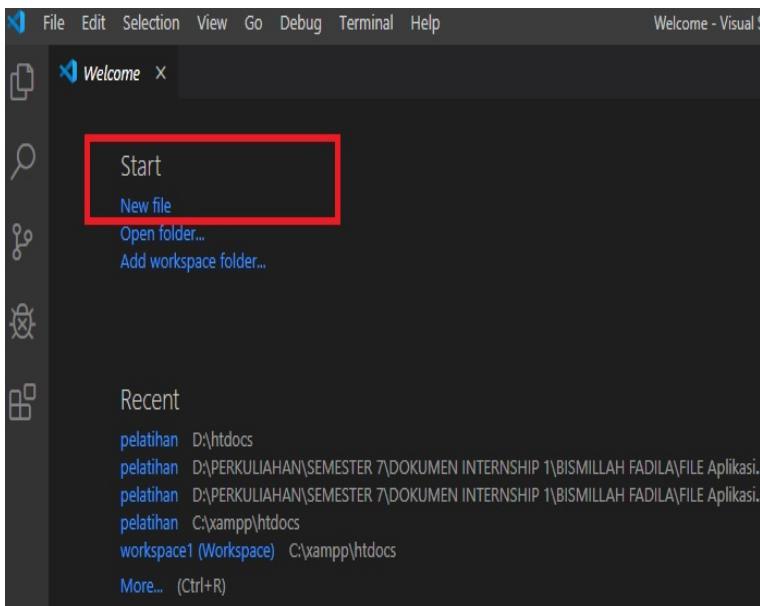
Gambar 3.74 Penggunaan Visual Studio Code 01

- (c) Silahkan klik icon tersebut kemudian akan muncul tampilan seperti ini:



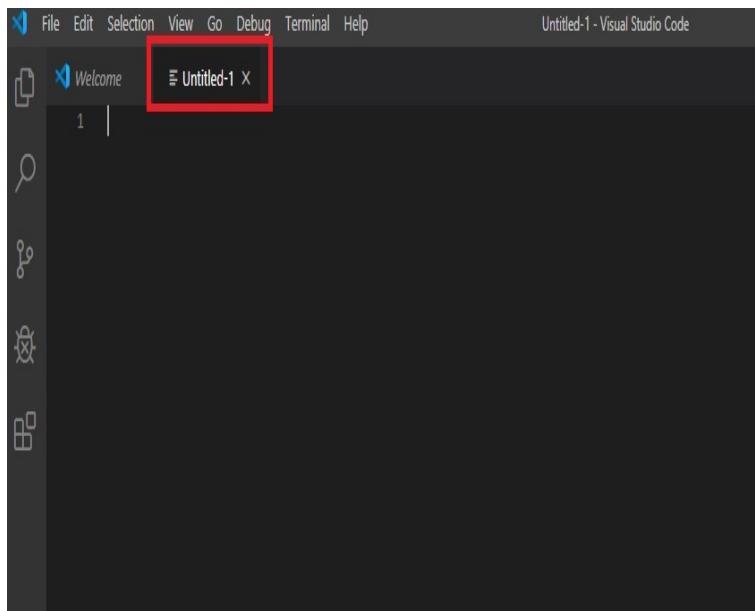
Gambar 3.75 Penggunaan Visual Studio Code 02

- (d) Tampilan yang anda lihat diatas merupakan tampilan awal ketika anda membuka Visual Studio Code apabila anda belum pernah menggunakannya. Namun, apabila anda sudah pernah menggunakan maka yang akan tampil ialah code terakhir yang anda kerjakan pada text editor tersebut.
- (e) Untuk memulai file baru silahkan anda klik hyperlink seperti yang dicontohkan pada gambar yaitu hyperlink *new start*.



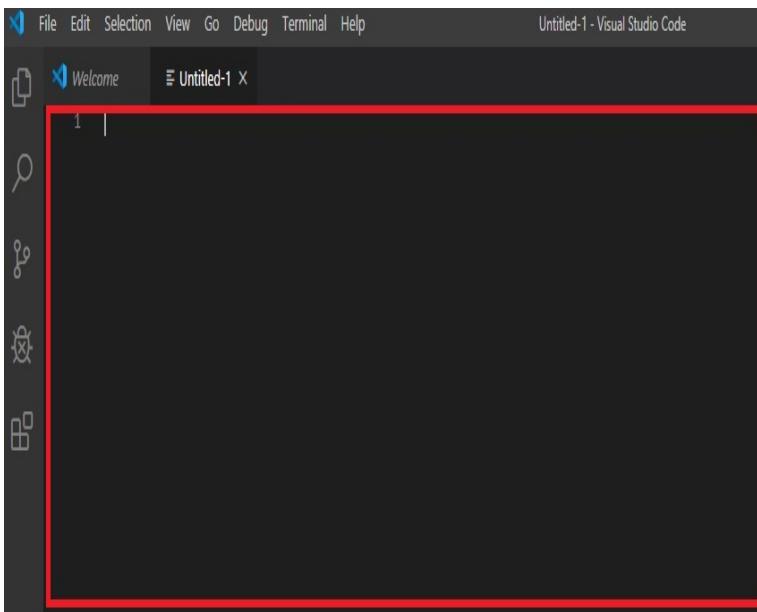
Gambar 3.76 Penggunaan Visual Studio Code 03

- (f) Apabila telah menekan hyperlink tersebut maka tampilannya akan nampak seperti ini:



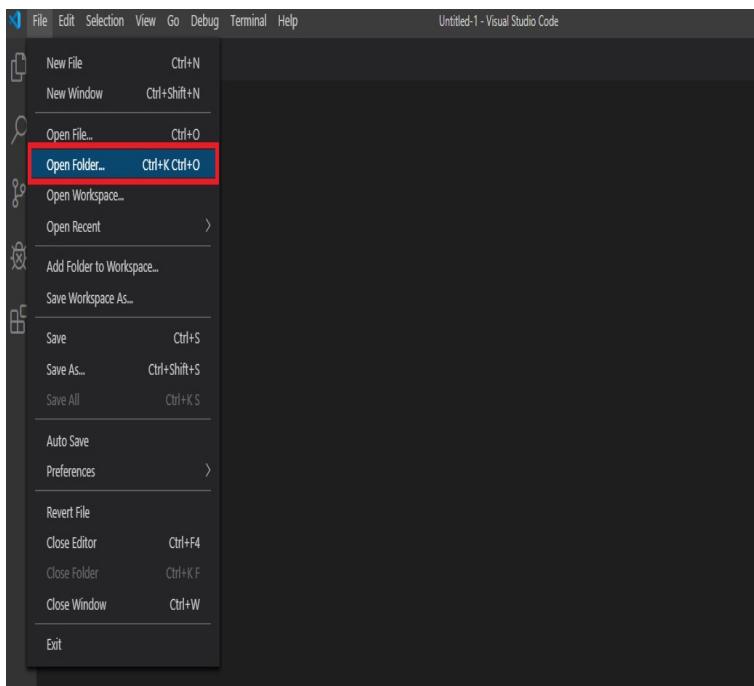
Gambar 3.77 Penggunaan Visual Studio Code 04

- (g) Untuk pengisian code dilakukan pada lembar kerja yang ditunjukkan oleh gambar. Untuk itu anda dapat memasukkan script code anda sesuai dengan kebutuhan pembangunan aplikasi:



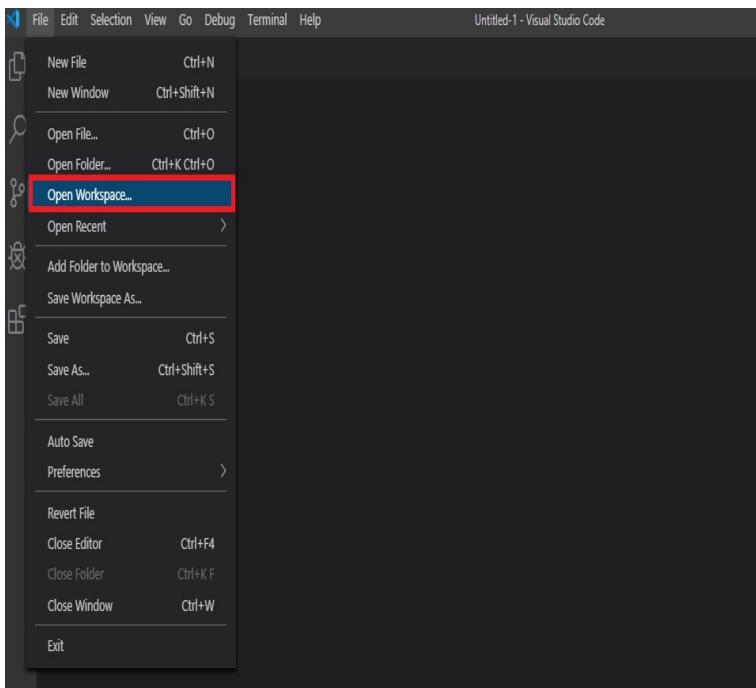
Gambar 3.78 Penggunaan Visual Studio Code 05

- (h) Langkah selanjutnya memperlihatkan cara membuka dan memasukkan folder file apabila anda telah menyediakannya sebelumnya:



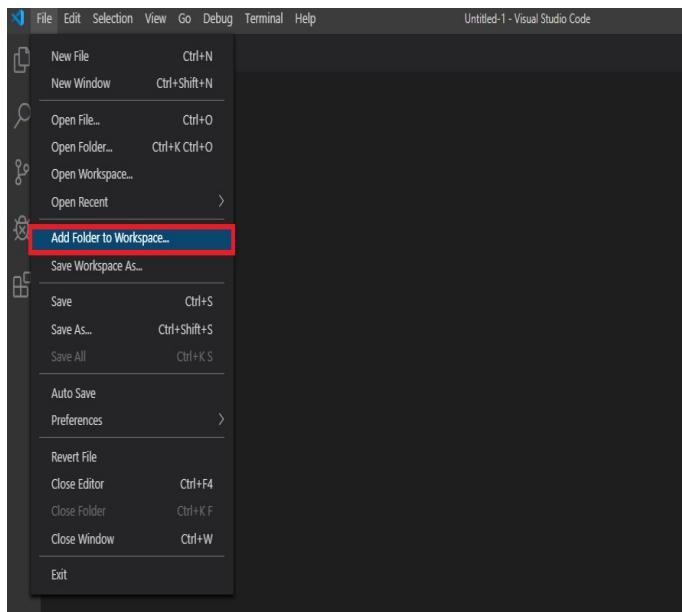
Gambar 3.79 Penggunaan Visual Studio Code 06

- (i) Selanjutnya anda juga bisa membuka dan memasukkan *workspace* atau lembar kerja apabila anda telah menyediakannya sebelumnya:



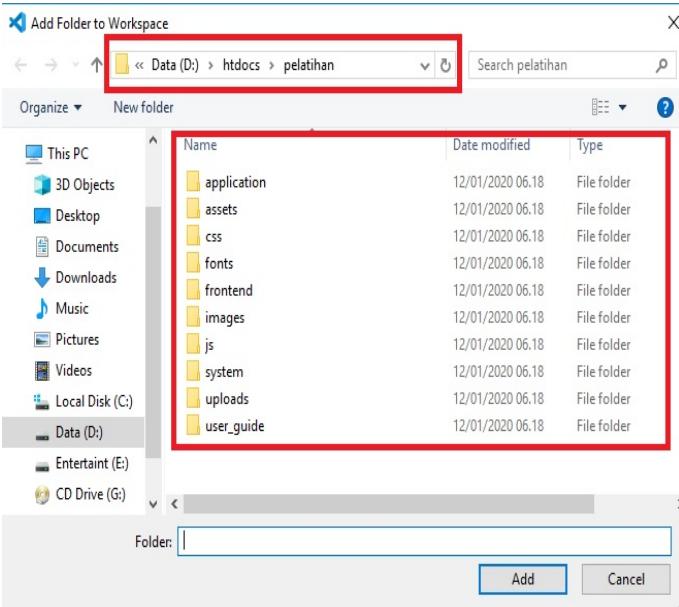
Gambar 3.80 Penggunaan Visual Studio Code 07

- (j) Anda juga bisa membuka dan membuat *workspace* atau lembar kerja sendiri apabila anda belum memiliki:



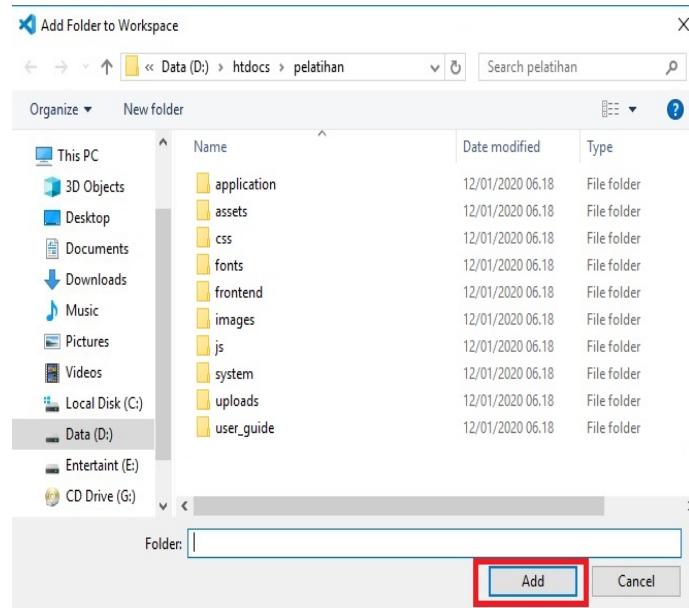
Gambar 3.81 Penggunaan Visual Studio Code 08

- (k) Untuk membuat workspace baru silahkan anda coba tambahkan folder dimana di dalamnya terdapat script. Apabila anda tidak memilikinya tidak apa-apanya cukup melihat panduan ini saja yang nantinya bisa dipakai pada kondisi yang lain
- (l) Pastikan folder tempat penyimpanan script anda sudah anda masukkan kedalam folder htdocs dari XAMPP yang ada di drive C agar dapat dijalankan nantinya:



Gambar 3.82 Penggunaan Visual Studio Code 09

- (m) Setelah itu silahkan klik button add untuk menambahkan folder script yang telah dipilih kedalam *workspace*:



Gambar 3.83 Penggunaan Visual Studio Code 10

- (n) Setelah mengikuti instruksi diatas, maka tampilannya akan nam-pak seperti gambar dimana foldernya telah tersimpan dan terbuka dalam *workspace* yang baru saja anda buat:

```

<li class="dropdown-item dropdown-hover">
    <i class="icon-book"></i>
    <a href="#">Master Data</a>
    <ul class="dropdown-menu">
        <li class="dropdown-header">Administrator</li>
        <li>
            <?php echo anchor('admin/Overview/tes15','User') ?>
        </li>
        <li>
            <?php echo anchor('admin/Overview/tes34','Pelatihan') ?>
        </li>
        <li>
            <?php echo anchor('admin/Overview/tes35','Regional') ?>
        </li>
        <li>
            <?php echo anchor('admin/Overview/tes36','Tempat Bekerja') ?>
        </li>
        <li>
            <?php echo anchor('admin/Overview/tes42','Kelompok') ?>
        </li>
        <li>
            <?php echo anchor('admin/Overview/tes37','Rekap Pelatihan') ?>
        </li>
        <li>
            <?php echo anchor('admin/Overview/tes38','Rekap Peserta') ?>
        </li>
    </ul>
</li>

```

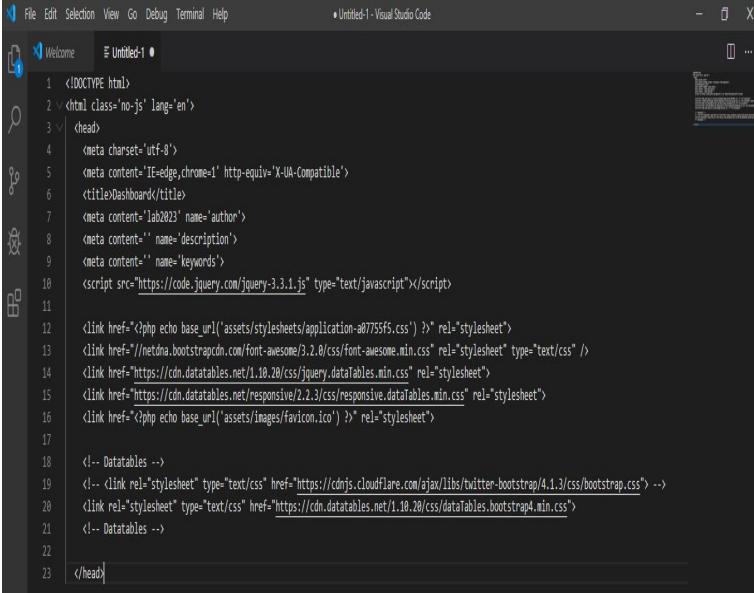
Gambar 3.84 Penggunaan Visual Studio Code 11

- (o) Baiklah untuk tahapan selanjutnya, anda bisa menekan CTRL+N pada keyboard untuk memulai file baru. Mengapa demikian? hal tersebut diperlukan untuk memudahkan anda dalam mengikuti panduan selanjutnya yang mana anda akan memasukkan script code.
- (p) Selanjutnya silahkan masukan script berikut kedalam file lembar kerja yang baru saja anda buat.

```

1 <!DOCTYPE html>
2 <html class='no-js' lang='en'>
3   <head>
4     <meta charset='utf-8'>
5     <meta content='IE=edge,chrome=1' http-equiv='X-UA-Compatible'>
6     <title>Dashboard </title>
7     <meta content='lab2023' name='author'>
8     <meta content='' name='description'>
9     <meta content='' name='keywords'>
10    <script src="https://code.jquery.com/jquery-3.3.1.js" type="text/javascript"></script>
11
12    <link href=<?php echo base_url('assets/stylesheets/application-a07755f5.css') ?>" rel="stylesheet">
13    <link href="//netdna.bootstrapcdn.com/font-awesome/3.2.0/css/font-awesome.min.css" rel="stylesheet" type="text/css" />
14    <link href="https://cdn.datatables.net/1.10.20/css/jquery.dataTables.min.css" rel="stylesheet">
15    <link href="https://cdn.datatables.net/responsive/2.2.3/css/responsive.dataTables.min.css" rel="stylesheet">
16    <link href=<?php echo base_url('assets/images/favicon.ico') ?>" rel="stylesheet">
17
18    <!-- Datatables -->
19    <!-- <link rel="stylesheet" type="text/css" href="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/4.1.3/css/bootstrap.css"> -->
20    <link rel="stylesheet" type="text/css" href="https://cdn.datatables.net/1.10.20/css/dataTables.bootstrap4.min.css">
21    <!-- Datatables -->
22
23  </head>
```

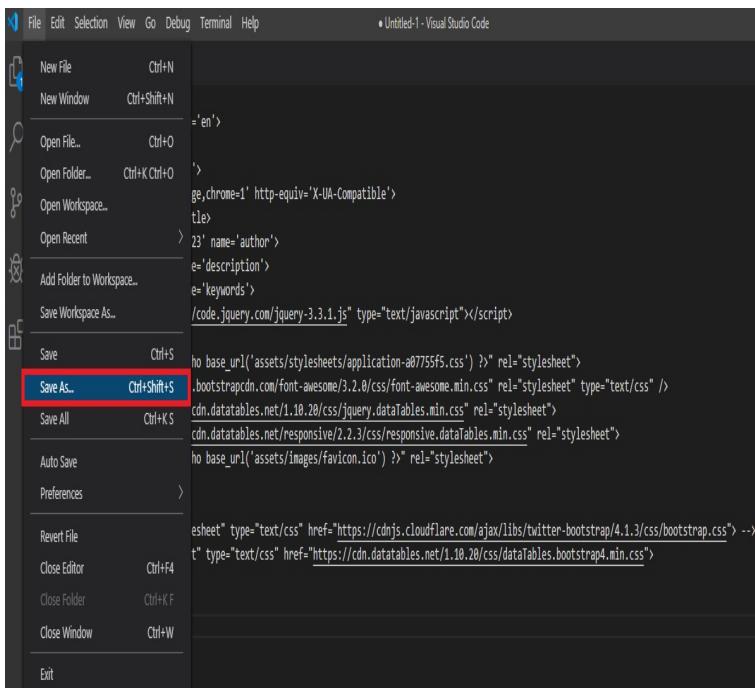
- (q) Script code diatas merupakan code *header* yang akan digunakan dalam pembangunan aplikasi. Disini hanya dijadikan contoh agar anda dapat lebih mengerti.
- (r) Setelah memasukkan code tersebut maka tampilannya akan nampak seperti gambar:



```
<!DOCTYPE html>
<html class="no-js" lang="en">
  <head>
    <meta charset="utf-8">
    <meta content="IE=edge,chrome=1" http-equiv="X-UA-Compatible" />
    <title>Dashboard</title>
    <meta content="lab2023" name="author">
    <meta content=" " name="description">
    <meta content=" " name="keywords">
    <script src="https://code.jquery.com/jquery-3.3.1.js" type="text/javascript"></script>
    <link href="php echo base_url('assets/stylesheets/application-a07755f5.css') ?&gt;" rel="stylesheet"&gt;
    &lt;link href="//netdna.bootstrapcdn.com/font-awesome/3.2.0/css/font-awesome.min.css" rel="stylesheet" type="text/css" /&gt;
    &lt;link href="https://cdn.datatables.net/1.10.20/css/jquery.dataTables.min.css" rel="stylesheet"&gt;
    &lt;link href="https://cdn.datatables.net/responsive/2.2.3/css/responsive.dataTables.min.css" rel="stylesheet"&gt;
    &lt;link href="<?php echo base_url('assets/images/favicon.ico') ?&gt;" rel="stylesheet"&gt;
    &lt;!-- Datatables --&gt;
    &lt;!-- &lt;link rel="stylesheet" type="text/css" href="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/4.1.3/css/bootstrap.css" --&gt;
    &lt;link rel="stylesheet" type="text/css" href="https://cdn.datatables.net/1.10.20/css/dataTables.bootstrap4.min.css" /&gt;
    &lt;!-- Datatables --&gt;
  &lt;/head&gt;</pre
```

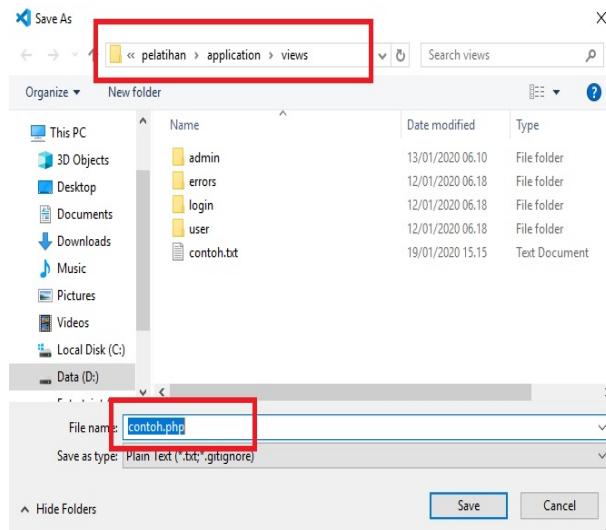
Gambar 3.85 Penggunaan Visual Studio Code 12

- (s) Script codenya berwarna putih dikarenakan belum disimpan
- (t) Selanjutnya silahkan simpan code sehingga nantinya dapat dijalankan pada eksekusi aplikasi
- (u) Silahkan menekan button Save As untuk melakukan aksi penyimpanan file



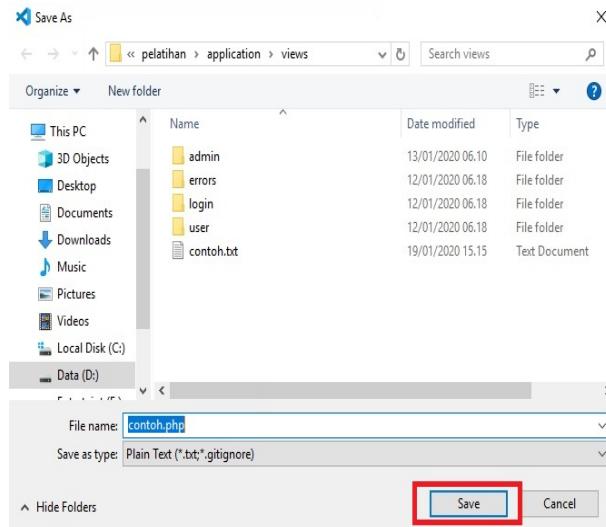
Gambar 3.86 Penggunaan Visual Studio Code 13

- (v) Silahkan sesuaikan penyimpanan file pada folder yang anda inginkan namun harus tetap berada pada folder htdocs dalam XAMPP agar dapat dieksekusi. Jangan lupa ekstensi file berupa PHP karena code ini merupakan code PHP:



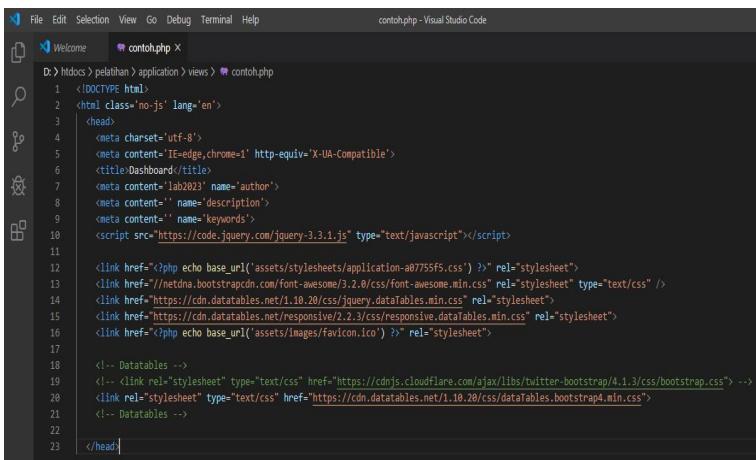
Gambar 3.87 Penggunaan Visual Studio Code 14

(w) Selanjutnya silahan menekan button save untuk menyimpan file:



Gambar 3.88 Penggunaan Visual Studio Code 15

(x) Setelah melakukan penyimpanan maka dapat dilihat codenya telah berubah warna yang menandakan bahwa penyimpanan telah sesuai dan code dapat dijalankan:



The screenshot shows the Visual Studio Code interface with the following details:

- Title Bar:** File, Edit, Selection, View, Go, Debug, Terminal, Help, contoh.php - Visual Studio Code
- File Explorer:** Shows a folder structure: D:\htdocs>pelatihan>application>views>contoh.php
- Code Editor:** Displays the following PHP code for a dashboard view:

```

1 <!DOCTYPE html>
2 <html class="no-js lang-'en'>
3   <head>
4     <meta charset='utf-8'>
5     <meta content='IE-edge,chrome=1' http-equiv='X-UA-Compatible'>
6     <title>Dashboard</title>
7     <meta content='lab2023' name='author'>
8     <meta content=' ' name='description'>
9     <meta content=' ' name='keywords'>
10    <script src="https://code.jquery.com/jquery-3.3.1.js" type="text/javascript"></script>
11
12    <link href=<?php echo base_url('assets/stylesheets/application-a07755f.css') ?>" rel="stylesheet">
13    <link href="https://netdna.bootstrapcdn.com/font-awesome/3.2.0/css/font-awesome.min.css" rel="stylesheet" type="text/css" />
14    <link href="https://cdn.datatables.net/1.10.20/css/jquery.dataTables.min.css" rel="stylesheet">
15    <link href="https://cdn.datatables.net/responsive/2.2.3/css/responsive.dataTables.min.css" rel="stylesheet">
16    <link href=<?php echo base_url('assets/images/favicon.ico') ?>" rel="stylesheet">
17
18    <!-- Datatables -->
19    <!-- <link rel="stylesheet" type="text/css" href="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/4.1.3/css/bootstrap.css"> -->
20    <link rel="stylesheet" type="text/css" href="https://cdn.datatables.net/1.10.20/css/dataTables.bootstrap4.min.css">
21    <!-- Datatables -->
22
23  </head>

```

Gambar 3.89 Penggunaan Visual Studio Code 16

- (y) Setelah mengikuti seluruh instruksi diatas, maka selesailah panduan dalam menggunakan text editor Visual Studio Code yang akan digunakan dalam pembangunan aplikasi.

3. Menggabungkan Bootstrap Dengan *Codigniter*

- **Langkah-Langkah Penggabungan Bootstrap dan Codeigniter**

BAB 4

PENJABARAN APLIKASI

4.1 Data Mining

Secara sederhana, data mining dapat diartikan sebagai proses mengekstrak atau menggali *knowledge* yang ada pada sekumpulan data. Informasi dan *knowledge* yang didapat tersebut dapat digunakan pada banyak bidang, seperti manajemen bisnis, pendidikan, kesehatan dan sebagainya.

Istilah data mining memiliki hakikat sebagai disiplin ilmu yang tujuan utamanya adalah untuk menemukan, menggali, atau menambang pengetahuan dari data atau informasi yang kita miliki. Proses menggali informasi dalam data mining melibatkan integrasi teknik dari berbagai disiplin ilmu, seperti teknologi *database* dan *data warehouse*, *statistik*, *machine learning*, komputasi dengan kinerja tinggi, *pattern recognition*, *neural network*, visualisasi data dan sebagainya.

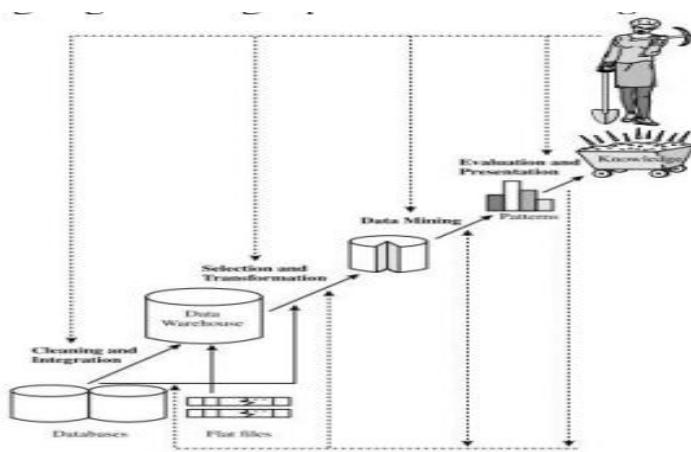
Data mining menggunakan pendekatan discovery-based dimana pencocokan pola (pattern matching) dan algoritma-algoritma yang lain digunakan untuk menentukan relasi-relasi kunci di dalam data yang dieksplorasi. Data mining (penambangan data), sesuai dengan namanya, berkonotasi sebagai pencarian informasi bisnis yang

berharga dari basis data yang sangat besar. Dengan tersedianya basis data dalam kualitas dan ukuran yang memadai, teknologi data mining memiliki kemampuan-kemampuan sebagai berikut:

- Mengotomatisasi prediksi trend sifat-sifat bisnis. Data mining mengotomatisasi proses pencarian informasi di dalam basis data yang besar.
- Mengotomatisasi penemuan pola-pola yang tidak diketahui sebelumnya. Tools data mining ”menyapu” basis data, kemudian mengidentifikasi pola-pola yang sebelumnya tersembunyi dalam satu sapuan. Contoh dari penemuan pola ini adalah analisis pada data penjualan ritel untuk mengidentifikasi produk-produk yang kelihatannya tidak berkaitan, yang seringkali dibeli secara bersamaan oleh customer.

4.1.1 Tahapan dalam *Data Mining*

Sebagai suatu rangkaian proses, data mining dapat dibagi menjadi beberapa tahap proses yang diilustrasikan pada Gambar 3 Tahap-tahap tersebut bersifat interaktif, pemakai terlibat langsung atau dengan perantaraan knowledge base.



Gambar 4.1 Tahapan Data Mining

Tahap-tahap data mining adalah sebagai berikut:

- Pembersihan data (data cleaning)

Pembersihan data merupakan proses menghilangkan noise dan data yang tidak konsisten atau data tidak relevan.

- Integrasi data (data integration)

Integrasi data merupakan penggabungan data dari berbagai database ke dalam satu database baru.

- Seleksi data (data selection)

Data yang ada pada database sering kali tidak semuanya dipakai, oleh karena itu hanya data yang sesuai untuk dianalisis yang akan diambil dari database.

- Transformasi data (data transformation)

Data diubah atau digabung ke dalam format yang sesuai untuk diproses dalam data mining.

- Proses mining

Merupakan suatu proses utama saat metode diterapkan untuk menemukan pengetahuan berharga dan tersembunyi dari data.

- Evaluasi pola (pattern evaluation)

Untuk mengidentifikasi pola-pola menarik ke dalam knowledge based yang ditemukan.

- Presentasi pengetahuan (knowledge presentation)

Merupakan visualisasi dan penyajian pengetahuan mengenai metode yang digunakan untuk memperoleh pengetahuan yang diperoleh pengguna.

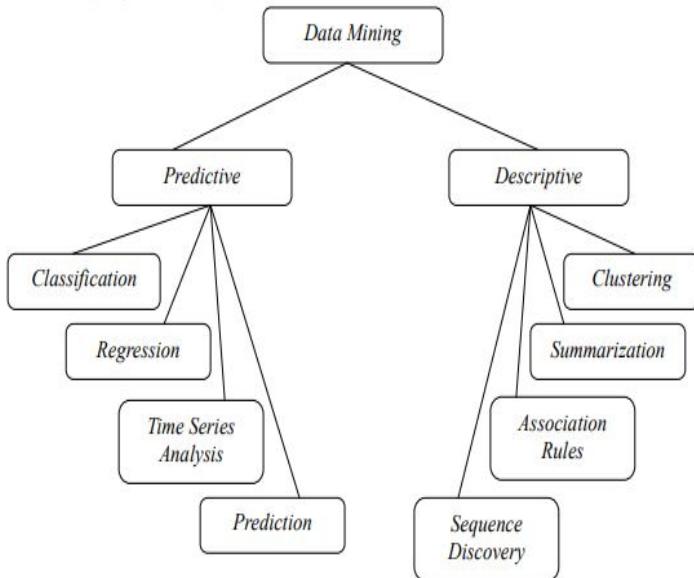
4.1.2 Teknik-Teknik Data mining

Data mining adalah serangkaian proses untuk menggali nilai tambah dari suatu kumpulan data berupa pengetahuan yang selama ini tidak diketahui secara manual.

Perlu diingat bahwa kata mining sendiri berarti usaha untuk mendapatkan sedikit data berharga dari sejumlah besar data dasar. Karena itu data mining sebenarnya

memiliki akar yang panjang dari bidang ilmu seperti kecerdasan buatan (artificial intelligent), machine learning, statistik dan basis data.

Teknik data mining biasanya terbagi dalam dua kategori, prediksi dan deskripsi. Teknik prediksi menggunakan data historis untuk menyimpulkan sesuatu tentang kejadian di masa depan. Sedangkan teknik deskripsi bertujuan untuk menemukan pola dalam data yang menyediakan beberapa informasi tentang hubungan interval yang tersembunyi.



Gambar 4.2 Teknik Data Mining

Menurut Kumar dan Saurabh, terdapat beberapa teknik yang digunakan dalam data mining, yaitu:

1. *Classification*

Klasifikasi adalah teknik yang paling umum diterapkan pada data mining. Pendekatan ini sering menggunakan keputusan pohon (decision tree) atau neural network berbasis algoritma klasifikasi. Proses klasifikasi data melibatkan learning dan klasifikasi.

Dalam belajar (learning) data pelatihan (training) dianalisis dengan algoritma klasifikasi. Dalam klasifikasi pengujian data dilakukan dengan menggunakan perkiraan akurasi dari aturan klasifikasi. Jika akurasi bisa diterima, maka aturan dapat diterapkan untuk data baru. Salah satu contoh yang mudah dan populer

adalah dengan decision tree yaitu salah satu metode klasifikasi yang paling populer karena mudah untuk diinterpretasi.

Decision tree adalah model prediksi menggunakan struktur pohon atau struktur berhirarki.

Decision tree adalah struktur flowchart yang menyerupai tree (pohon), dimana setiap simpul internal menandakan suatu tes pada atribut, setiap cabang merepresentasikan hasil tes, dan simpul daun merepresentasikan kelas atau distribusi kelas. Alur pada decision tree di telusuri dari simpul akar ke simpul daun yang memegang prediksi kelas untuk contoh tersebut. Decision tree mudah untuk dikonversi ke aturan klasifikasi (*classification rules*).

2. *Clustering*

Clustering bisa dikatakan sebagai identifikasi kelas objek yang memiliki kemiripan. Dengan menggunakan teknik clustering kita bisa lebih lanjut mengidentifikasi kepadatan dan jarak daerah dalam objek ruang dan dapat menemukan secara keseluruhan pola distribusi dan korelasi antara atribut. Pendekatan klasifikasi secara efektif juga dapat digunakan untuk membedakan kelompok atau kelas objek.

3. Prediksi

Teknik regresi dapat disesuaikan untuk predikasi. Analisis regresi dapat digunakan untuk model hubungan antara satu atau lebih independent variables dan dependent variables.

Dalam data mining independent variabel adalah atribut-atribut yang sudah dikenal dan respon variabel apa yang kita inginkan untuk diprediksi. Akan tetapi, banyak masalah di dunia nyata bukan prediksi yang mudah. Karena itu, teknik kompleks (seperti: logistic regression, decision trees atau pohon keputusan, neural nets atau jaringan syaraf) mungkin akan diperlukan untuk memprediksi nilai.

Model yang berjenis sama sering dapat digunakan untuk regresi dan klasifikasi. Misalnya, CART (Classification and Regression Trees) yaitu algoritma pohon keputusan yang dapat digunakan untuk membangun kedua pohon klasifikasi dan pohon regresi. Jaringan saraf juga dapat menciptakan kedua model klasifikasi dan regresi.

4. Association rule

Digunakan untuk mengenali kelakuan dari kejadian-kejadian khusus atau proses dimana link asosiasi muncul pada setiap kejadian. Contoh dari aturan assosiatif dari analisa pembelian di suatu pasar swalayan adalah bisa diketahui berapa besar kemungkinan seorang pelanggan membeli roti bersamaan dengan susu. Dengan pengetahuan tersebut pemilik pasar swalayan dapat mengatur penempatan barangnya atau merancang kampanye pemasaran dengan memakai kupon diskon untuk kombinasi barang tertentu.

Penting tidaknya suatu aturan assosiatif dapat diketahui dengan dua parameter, support yaitu prosentasi kombinasi atribut tersebut dalam basisdata dan confidence yaitu kuatnya hubungan antar atribut dalam aturan assosiatif. Motivasi awal pencarian association rule berasal dari keinginan untuk menganalisa data transaksi supermarket, ditinjau dari perilaku customer dalam membeli produk. Association rule ini menjelaskan seberapa sering suatu produk dibeli secara bersamaan.

Sebagai contoh, association rule beer \Rightarrow diaper 80persen menunjukkan bahwa empat dari lima customer yang membeli beer juga membeli diaper. Dalam suatu association rule X \Rightarrow Y, X disebut dengan antecedent dan Y disebut dengan consequent rule.

5. Neural network

Jaringan saraf adalah seperangkat unit penghubung input dan output dimana setiap koneksinya memiliki bobot. Selama fase learning, jaringan belajar dengan menyesuaikan bobot sehingga dapat memprediksi kelas yang benar label dari setiap input.

Jaringan saraf memiliki kemampuan yang luar biasa untuk memperoleh arti dari data yang rumit atau tidak tepat dan dapat digunakan untuk mengambil pola-pola serta mendeteksi tren yang sangat kompleks untuk diperhatikan baik oleh manusia atau teknik komputer lain.

Jaringan saraf sangat baik untuk mengidentifikasi pola atau tren pada data dan sangat cocok untuk melakukan prediksi serta memprediksi kebutuhan.

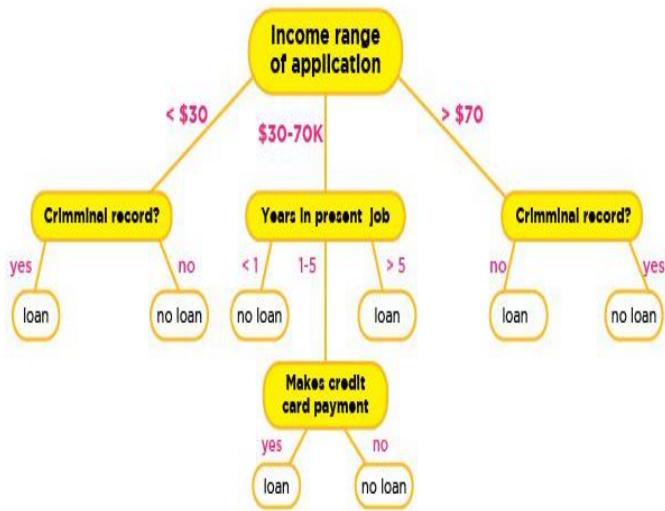
6. Decision trees

Decision trees atau pohon keputusan adalah struktur tree-shaped yang mewakili set keputusan. Keputusan ini menghasilkan aturan untuk klasifikasi sebuah kumpulan data. Metode pohon keputusan diantaranya yaitu Classification and

regression trees (CART) dan Chi Square Automatic Interaction Detection (CHAID).

Decision tree juga berguna untuk mengeksplorasi data, menemukan hubungan tersembunyi antara sejumlah calon variabel input dengan sebuah variabel target. Decision tree memadukan antara eksplorasi data dan pemodelan, sehingga sangat bagus sebagai langkah awal dalam proses pemodelan bahkan ketika dijadikan sebagai model akhir dari beberapa teknik lain.

Dalam beberapa aplikasi, akurasi dari sebuah klasifikasi atau prediksi adalah satu-satunya hal yang ditonjolkan dalam metode ini, misalnya sebuah perusahaan direct mail membuat sebuah model yang akurat untuk memprediksi anggota mana yang berpotensi untuk merespon permintaan, tanpa memperhatikan bagaimana atau mengapa model tersebut bekerja.



Gambar 4.3 Decision Tree

Kelebihan lain dari metode ini adalah mampu mengeliminasi perhitungan atau data-data yang kiranya tidak diperlukan. Sebab, sampel yang ada biasanya hanya diuji berdasarkan kriteria atau kelas tertentu saja.

Meski memiliki banyak kelebihan, namun bukan berarti metode ini tidak memiliki kekurangan. Decision tree ini bisa terjadi overlap, terutama ketika kelas dan kriteria yang digunakan sangat banyak tentu saja dapat meningkatkan waktu

pengambilan keputusan sesuai dengan jumlah memori yang dibutuhkan.

Dalam hal akumulasi, decision tree juga seringkali mengalami kendala eror terutama dalam jumlah besar. Selain itu, terdapat pula kesulitan dalam mendesain decision tree yang optimal. Apalagi mengingat kualitas keputusan yang didapatkan dari metode decision tree sangat tergantung pada bagaimana pohon tersebut didesain.

Terlepas dari kekurangan dan kelebihan dari decision tree, metode ini banyak digunakan lebih lanjut dalam berbagai pengolahan data. Mulai dari data mining dan juga machine learning. Dalam dunia kerja, decision tree sendiri sangat berguna untuk penilaian credit scoring. Jika anda pernah mengajukan kredit yang diproses secara instan, nah anda sudah mempunyai pengalaman dari decision tree.

7. Nearest Neighbor Method

Teknik yang mengklasifikasikan setiap record dalam sebuah kumpulan data berdasarkan sebuah kombinasi suatu kelas k record yang sama dalam sebuah kumpulan data historis (dimana k lebih besar atau sama dengan 1). Terkadang disebut juga dengan teknik K-Nearest Neighbor.

4.1.3 Sepuluh Top Algoritma Data Mining

Dari 18 nominasi algoritma yang dibagi dalam 10 topik, telah diumumkan 10 Algoritma tersebut. Dari 18 nominasi tersebut dilakukan voting oleh anggota program *committee KDD-06, ICDM06, SDM06*, pemenang *ACM KDD Innovation Award*, dan pemenang *IEEE ICDM Research Contribution Award* yang kemudian diperoleh ranking 10 algoritma teratas. Berikut ini adalah hasilnya (Nomor tidak menunjukkan peringkatnya):

1. Pohon Keputusan (*Decision Tree*)

metode klasifikasi dan prediksi yang sangat kuat dan terkenal. Metode pohon keputusan mengubah fakta yang sangat besar menjadi pohon keputusan yang merepresentasikan aturan. Aturan dapat dengan mudah dipahami dengan bahasa alami. Aturan ini juga dapat diekspresikan dalam bentuk bahasa basis data seperti *SQL* untuk mencari record pada kategori tertentu. Pohon keputusan juga berguna untuk mengeksplorasi data, menemukan hubungan tersembunyi antara sejumlah calon variabel input dengan sebuah variabel target. Karena pohon keputusan memadukan antara eksplorasi data dan pemodelan, pohon keputusan ini sangat bagus sebagai langkah awal dalam proses pemodelan bahkan ketika

dijadikan sebagai model akhir dari beberapa teknik lain.

Contoh Perhitungan Decision Tree dengan Algoritma C45

Untuk menentukan bermain tenis atau tidak, kriteria yang diperlukan meliputi:

- (a) Cuaca
- (b) Angin
- (c) Kelembaban
- (d) Temperatur udara

Salah satu atribut merupakan data solusi per item data yang disebut target atribut $-i$, misalnya atribut play dengan nilai main atau tidak main.

Atribut memiliki nilai-nilai yang dinamakan instance. Misalkan atribut Cuaca memiliki instance $-i$ cerah, berawan, dan hujan.

No	Outlook	Temperature	Humidity	Windy	Play
1	Sunny	Hot	High	FALSE	No
2	Sunny	Hot	High	TRUE	No
3	Cloudy	Hot	High	FALSE	Yes
4	Rainy	Mild	High	FALSE	Yes
5	Rainy	Cool	Normal	FALSE	Yes
6	Rainy	Cool	Normal	TRUE	Yes
7	Cloudy	Cool	Normal	TRUE	Yes
8	Sunny	Mild	High	FALSE	No
9	Sunny	Cool	Normal	FALSE	Yes
10	Rainy	Mild	Normal	FALSE	Yes
11	Sunny	Mild	Normal	TRUE	Yes
12	Cloudy	Mild	High	TRUE	Yes
13	Cloudy	Hot	Normal	FALSE	Yes
14	Rainy	Mild	High	TRUE	No

Gambar 4.4 Contoh C45

Berdasarkan tabel diatas akan dibuat tabel keputusan untuk menentukan main tenis atau tidak dengan melihat keadaan Outlook (cuaca), Temperatur, Humidity (kelembaban), dan windy (keadaan angin).

Algoritma secara umum:

- Pilih atribut sebagai akar
- Buat cabang untuk tiap2 nilai
- Bagi kasus dalam cabang
- Ulangi proses utk setiap cabang sampai semua kasus pada cabang memiliki kelas yang sama
- Memilih atribut berdasarkan nilai gain tertinggi dari atribut-atribut yang ada.

Perhitungan Gain:

$$\text{Gain}(S, A) = \text{Entropy}(S) - \sum_{i=1}^n p_i * \text{Entropy}(S_i)$$

Gambar 4.5 Perhitungan Gain

Keterangan:

- S : himpunan
- A : atribut
- n : jumlah partisi atribut A
- — Si — : jumlah kasus pada partisi ke-i
- — S — : jumlah kasus dalam S

Menghitung Nilai Entropy:

$$\text{Entropy } (S) = \sum_{i=1}^n -p_i * \log_2 p_i$$

Gambar 4.6 Perhitungan Entropi

Keterangan:

- S : himpunan kasus
- A : fitur
- n : jumlah partisi S
- p_i : proporsi dari S_i terhadap S
- — S — : jumlah kasus dalam S

Perincian algoritma (langkah 1):

- Menghitung jumlah kasus seluruhnya, jumlah berkeputusan Yes maupun No.
- Menghitung Entropy dari semua kasus yg terbagi berdasarkan atribut Outlook, Temperature, Humidity, Windy.
- Lakukan penghitungan Gain utk setiap atributnya

Node			Jumlah Kasus (S_1)	Tidak (S_1)	Ya (S_2)	Entropy	Gain
1	TOTAL		14	4	10	0,863120569	
	OUTLOOK						0,258521037
		CLOUDY	4	0	4	0	
		RAINY	5	1	4	0,721928095	
		SUNNY	5	3	2	0,970950594	
	TEMPERATURE						0,183850925
		COOL	4	0	4	0	
		HOT	4	2	2	1	
		MILD	6	2	4	0,918295834	
	HUMIDITY						0,370506501
		HIGH	7	4	3	0,985228136	
		NORMAL	7	0	7	0	
	WINDY						0,005977711
		FALSE	8	2	6	0,811278124	
		TRUE	6	2	4	0,918295834	

Gambar 4.7 Perhitungan

Node			Jumlah Kasus (S_1)	Tidak (S_1)	Ya (S_2)	Entropy	Gain
1	TOTAL		14	4	10	0,863120569	
...

Gambar 4.8 Perhitungan Entropi Total

$$\text{Entropy } (S) = \sum_{i=1}^n - p_i * \log_2 p_i$$

$$\begin{aligned} \text{Entropy (Total)} = & - 4/14 * \log_2 (4/14) + \\ & - 10/14 * \log_2 (10/14) \end{aligned}$$

Gambar 4.9 Rumus Entropi Total

Menghitung gain pada baris Outlook

$$\begin{aligned}
 & \text{Gain (Total, Outlook)} \\
 & = \text{Entropy}(\text{Total}) - \sum_{i=1}^n \frac{\text{Outlook}_i}{\text{Total}} * \text{Entropy}(\text{Outlook}_i) \\
 & = 0,863120569 - ((4/14 * 0) + (5/14 * 0,72) + (5/14 * 0,97)) \\
 & = 0,258521037
 \end{aligned}$$

Gambar 4.10 Menghitung gain

Lakukan Hitung Gain untuk temperature, humidity dan windy

Seperti yg terlihat pd tabel, diperoleh bhw atribut dgn Gain tertinggi adalah Humidity -i 0,37

Maka Humidity menjadi node akar

Humidity memiliki dua nilai yaitu High dan Normal

Humidity -i Normal sdh mengklasifikasikan kasus menjadi 1 yaitu keputusannya yes

Untuk humidity -i High msh perlu dilakukan perhitungan lagi (karena masih terdapat yes dan no)

Pohon Keputusan Node 1

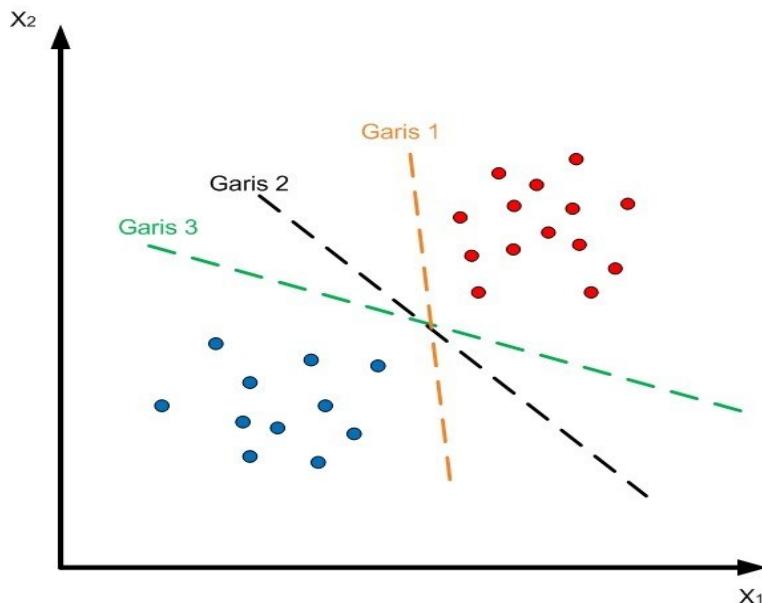
2. K-Means

K-means merupakan salah satu metode *clustering non hirarki* yang berusaha mempartisi data yang ada ke dalam bentuk satu atau lebih *cluster*. Metode ini mempartisi data ke dalam *cluster* sehingga data yang memiliki karakteristik yang sama dikelompokkan ke dalam satu *cluster* yang sama dan data yang mempunyai karakteristik yang berbeda di kelompokan ke dalam *cluster* yang lain.

3. Support Vector Machines

SVM adalah metode *machine learning* yang bekerja atas prinsip *Structural Risk Minimization (SRM)* dengan tujuan menemukan hyperplane terbaik yang memisahkan dua buah class pada input space. *SVM* merupakan sistem pembelajaran yang menggunakan ruang hipotesis berupa fungsi-fungsi linier dalam sebuah ruang fitur (feature space) berdimensi tinggi, dilatih dengan algoritma pembelajaran yang didasarkan pada teori optimasi dengan mengimplementasikan *learning bias* yang berasal dari teori pembelajaran statistik.

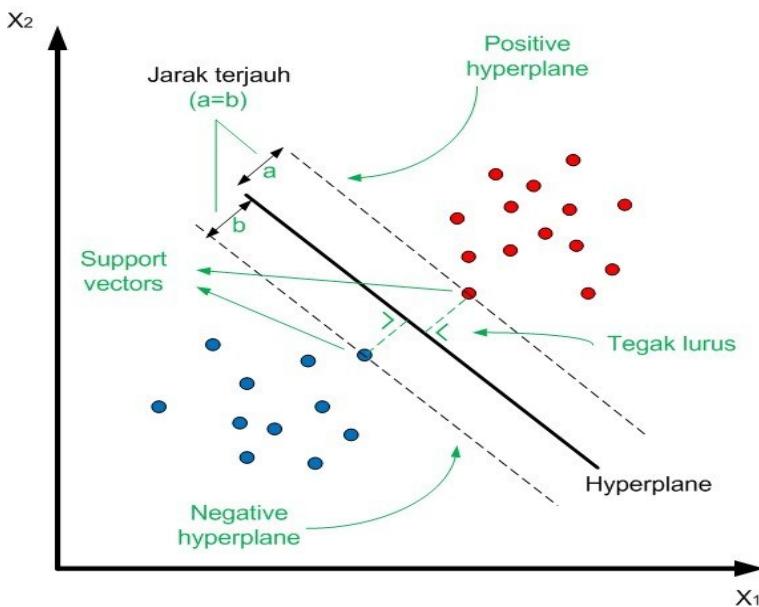
Cara kerja SVM bisa dilihat melalui ilustrasi yang sangat sederhana sebagai berikut:



Gambar 4.11 Ilustrasi SVM.

Ilustrasi SVM. Terlihat ada 3 garis pembatas (1, 2, dan 3). Masing-masing garis ini dibentuk oleh 2 titik yang terdekat dari garisnya.

Bisa dilihat bahwa terdapat 2 kelompok data (pembagiannya sudah kita ketahui, oleh karena itu disebut klasifikasi, bukan clustering). Kemudian, tugas SVM adalah membagi 2 kelompok ini sebaik mungkin



Gambar 4.12 Ilustrasi SVM.

Ilustrasi bagaimana menghitung jarak antara titik terluar di masing-masing kelompok dengan hyperplane.

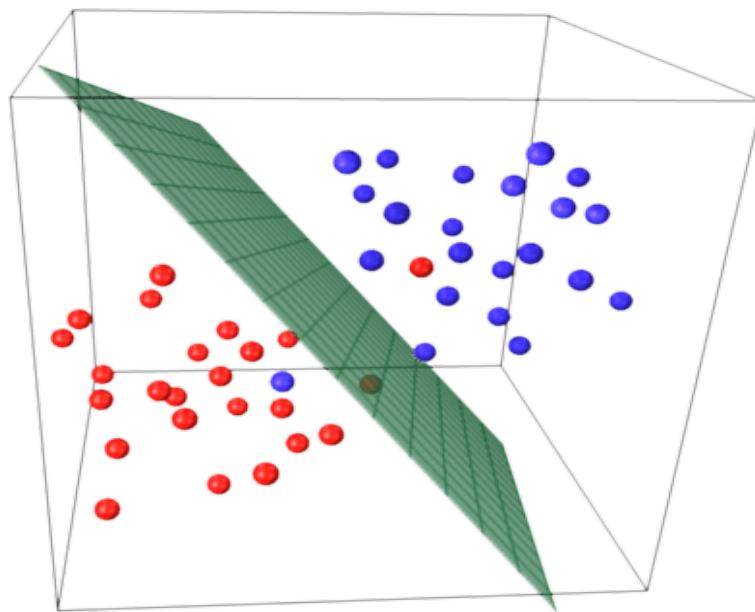
Perlu diperhatikan bahwa 2 titik terluar ini (paling luar dari kelompoknya dan paling dekat dengan hyperplane) harus tegak lurus terhadap hyperplane nya. Dua titik ini disebut dengan support vectors.

Dua titik yang menjadi patokan hyperplane disebut dengan support vectors. Menariknya, titik-titik lain di belakang dua garis ini tidak berkontribusi apa-apa terhadap hasil SVM, karena sesuai namanya, baik tidaknya hasil SVM tergantung dari support vectors-nya.

Sebagai tambahan, penamaan positive hyperplane atau negative hyperplane itu bebas. Jadi garis atas bisa saja kita sebut sebagai negative, dan garis bawah menjadi positive.

Hal lain yang harus dipahami bahwa penamaan hyperplane sebagai garis pembatas adalah penamaan yang banyak dikenal oleh data scientist di luar sana. Pembaca boleh saja menyebutnya hyperline (untuk kasus 2 dimensi, karena memang hanya tampak sebagai sebuah garis saja), atau konsisten dengan penamaan hyperplane (karena untuk 3D atau lebih, hyperplane memang pembatas

yang berupa a plane (satu sisi pembatas) dan bukan hanya sekedar garis. Jika 3D kurang lebih akan tampak seperti ini:



Gambar 4.13 Ilustrasi SVM.

Keterangan:

Ilustrasi SVM. Terlihat ada 3 garis pembatas (1, 2, dan 3).

Masing-masing garis ini dibentuk oleh 2 titik terdekat yang terdekat dari garisnya.

Metode Support Vector Machine memiliki beberapa keuntungan yaitu:

- (a) Generalisasi

Generalisasi didefinisikan sebagai kemampuan suatu metode untuk mengklasifikasi suatu pattern atau pola, yang tidak termasuk data yang digunakan dalam fase pembelajaran metode itu.

- (b) Curse of dimensionality

Curse of dimensionality didefinisikan sebagai masalah yang dihadapi suatu metode pattern recognition dalam mengestimasikan parameter dikarenakan jumlah sampel data yang relatif lebih sedikit dibandingkan dengan dimensional ruang vektor tersebut.

(c) Feasibility

SVM dapat diimplementasikan relatif lebih mudah, karena proses penentuan support vector dapat dirumuskan dalam Quadratic Programming (QP) problem

Adapun kerugian dari metode Support Vector Machine adalah sebagai berikut:

- (a) Sulit dipakai pada problem berskala besar. Dalam hal ini dimaksudkan dengan jumlah sampel yang diolah.
- (b) SVM secara teoritik dikembangkan untuk problem klasifikasi dengan dua kelas. Namun dewasa ini SVM telah dimodifikasi agar dapat menyelesaikan masalah dengan lebih dari dua kelas

4. Apriori

Algoritma Apriori adalah algoritma paling terkenal untuk menemukan pola frekuensi tinggi. Pola frekuensi tinggi adalah pola-pola item di dalam suatu database yang memiliki frekuensi atau support di atas ambang batas tertentu yang disebut dengan istilah minimum support.

Pola frekuensi tinggi ini digunakan untuk menyusun aturan assosiatif dan juga beberapa teknik data mining lainnya.

Walaupun akhir-akhir ini dikembangkan banyak algoritma yang lebih efisien dari Apriori seperti FP-growth, LCM dsb, tetapi Apriori tetap menjadi algoritma yang paling banyak diimplementasikan dalam produk komersial untuk data mining karena dianggap algoritma yang paling mapan.

Algoritma Apriori dibagi menjadi beberapa tahap yang disebut iterasi atau pass. Tiap iterasi menghasilkan pola frekuensi tinggi dengan panjang yang sama dimulai dari pass pertama yang menghasilkan pola frekuensi tinggi dengan panjang satu.

Di iterasi pertama ini, support dari setiap item dihitung dengan men-scan database. Setelah support dari setiap item didapat, item yang memiliki support diatas minimum support dipilih sebagai pola frekuensi tinggi dengan panjang 1 atau sering disingkat 1-itemset. Singkatan k-itemset berarti satu set yang terdiri dari k item.

Iterasi kedua menghasilkan 2-itemset yang tiap set-nya memiliki dua item. Pertama dibuat kandidat 2-itemset dari kombinasi semua 1-itemset. Lalu untuk tiap kandidat 2-itemset ini dihitung support-nya dengan men-scan database.

Support disini artinya jumlah transaksi dalam database yang mengandung kedua item dalam kandidat 2-itemset. Setelah support dari semua kandidat 2-itemset didapatkan, kandidat 2-itemset yang memenuhi syarat minimum support dapat ditetapkan sebagai 2-itemset yang juga merupakan pola frekuensi tinggi dengan panjang 2.

Untuk selanjutnya pada iterasi ke-k dapat dibagi lagi menjadi beberapa bagian :

- Pembentukan kandidat itemset, Kandidat k-itemset dibentuk dari kombinasi $(k-1)$ -itemset yang didapat dari iterasi sebelumnya. Satu ciri dari algoritma Apriori adalah adanya pemangkasan kandidat k-itemset yang subset-nya yang berisi $k-1$ item tidak termasuk dalam pola frekuensi tinggi dengan panjang $k-1$.
- Penghitungan support dari tiap kandidat k-itemset. Support dari tiap kandidat k-itemset didapat dengan men-scan database untuk menghitung jumlah transaksi yang memuat semua item di dalam kandidat k-itemset tsb. Ini adalah juga ciri dari algoritme Apriori dimana diperlukan penghitungan dengan scan seluruh database sebanyak k-itemset terpanjang.
- Tetapkan pola frekuensi tinggi. Pola frekuensi tinggi yang memuat k item atau k-itemset ditetapkan dari kandidat k-itemset yang support-nya lebih besar dari minimum support.
- Bila tidak didapat pola frekuensi tinggi baru maka seluruh proses dihentikan. Bila tidak, maka k ditambah satu dan kembali ke bagian 1.

Pseudocode dari algoritma Apriori dapat dilihat di Gambar berikut :

```

 $L_1 := \{ \text{large 1-itemsets} \};$ 
 $k := 2; \quad \quad // k \text{ represents the pass number}$ 
while ( $L_{k-1} \neq \emptyset$ ) do
begin
     $C_k := \text{New candidates of size } k \text{ generated from } L_{k-1}; (\underline{\text{apriori\_gen}})$ 
    forall transactions  $t \in \mathcal{D}$  do
        Increment the count of all candidates in  $C_k$  that are contained in  $t$ ;
     $L_k := \text{All candidates in } C_k \text{ with minimum support};$ 
     $k := k + 1;$ 
end
Answer :=  $\bigcup_k L_k;$ 

```

Gambar 4.14 Pseudocode dari algoritma Apriori

Sedangkan pseudocode dari pembentukan kandidat itemset bersama pemangkasannya diberikan di Gambar berikut :

-(1) Join Step

```

insert into candidate  $k$ -itemset
select  $p.item_1, p.item_2, \dots, p.item_{k-1}$ 
from large  $(k-1)$ -itemset  $p$ , large  $(k-1)$ -itemset  $q$ 
where  $p.item_1 = q.item_1, \dots, p.item_{k-2} = q.item_{k-2}, p.item_{k-1} < q.item_{k-1};$ 

```

(2) Prune Step

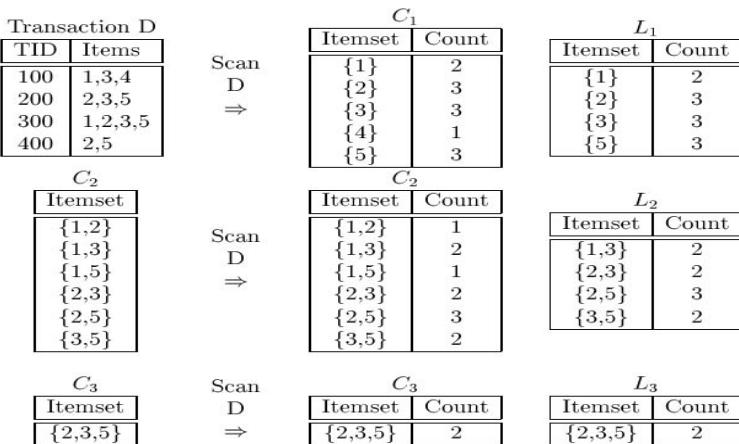
```

forall itemsets  $c \in$  candidate  $k$ -itemset do
    forall  $(k-1)$ -subsets  $s$  of  $c$  do
        if ( $s \notin$  large  $(k-1)$ -itemset) then
            delete  $c$  from candidate  $k$ -itemset;

```

Gambar 4.15 pseudocode dari pembentukan kandidat itemset

Satu contoh dari penerapan algoritma Apriori diilustrasikan di Gambar berikut :



Gambar 4.16 Contoh Algoritma Apriori

Di sini minimum support adalah 50persen atau minimal support-nya adalah 2. Pada iterasi pertama, item yang support-nya atau count-nya dibawah 2 dileliminasi dari 1-itemset L1. Kemudian kandidat 2-itemset C2 dari iterasi kedua dibentuk dari cross product item-item yang ada di L1. Setelah kandidat 2-itemset itu dihitung dari database, ditetapkan 2-itemset L2. Proses serupa berulang di iterasi ketiga, tetapi perhatikan bahwa selain 2,3,5 yang menjadi kandidat 3-itemset C3 sebenarnya ada juga itemset 1,2,3 dan 1,3,5 yang dapat diperoleh dari kombinasi item-item di L2, tetapi kedua itemset itu dipangkas karena 2,3 dan 1,5 tidak ada di L2. Proses ini berulang sampai tidak ada lagi kandidat baru yang dapat dihasilkan di iterasi ke 4.

Dalam contoh ini bisa dilihat bahwa Apriori dapat mengurangi jumlah kandidat yang harus dihitung support-nya dengan pemangkasan. Misalnya kandidat 3-itemset dapat dikurangi dari 3 menjadi 1 saja. Pengurangan jumlah kandidat ini merupakan sebab utama peningkatan performa Apriori.

Tetapi di lain pihak Apriori memiliki kelemahan karena harus melakukan scan database setiap kali iterasi, sehingga waktu yang diperlukan bertambah dengan makin banyak iterasi. Masalah ini yang dipecahkan oleh algoritma-algoritma baru seperti FP-growth.

5. Expectation Maximisation Algorithm

Expectation Maximisation Algorithm (EM Algorithm) adalah algoritma yang sering digunakan untuk menemukan nilai estimasi *Maximum Likelihood (ML)*

dari parameter dalam sebuah model *probabilistic*, dimana model juga tergantung pada latent variabel yang belum diketahui. Dalam algoritma ini, ada dua hal yang dilakukan secara bergantian yaitu *E step* yang menghitung nilai ekspektasi dari *likelihood* termasuk laten variabel seolah-olah seperti mereka ada, dan *M step* menghitung nilai estimasi ML dari parameter dengan memaksimalkan nilai ekspektasi dari *likelihood* yang ditemukan pada *E step*

6. *PageRank*

PageRank adalah sebuah algoritma yang telah dipatenkan yang berfungsi menentukan situs web mana yang lebih penting/populer. *PageRank* merupakan salah satu fitur utama mesin pencari *Google* dan diciptakan oleh pendirinya, *Larry Page* dan *Sergey Brin* yang merupakan mahasiswa *Ph.D. Universitas Stanford*.

7. *k-Nearest Neighbors*

K-Nearest Neighbor (KNN) adalah suatu metode yang menggunakan *algoritma-supervised* dimana hasil dari *query instance* yang baru diklasifikasi berdasarkan mayoritas dari kategori pada KNN. Tujuan dari algoritma ini adalah mengklasifikasikan obyek baru berdasarkan atribut dan training sample. Algoritma metode KNN sangatlah sederhana, bekerja berdasarkan jarak terpendek dari *query instance* ke *training sample* untuk menentukan KNN-nya. Training sample diproyeksikan ke ruang berdimensi banyak, dimana masing-masing dimensi merepresentasikan fitur dari data. Ruang ini dibagi menjadi bagian-bagian berdasarkan klasifikasi *training sample*.

8. *Nave Bayes*

Nave Bayes merupakan pengklasifikasian dengan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dikenal sebagai teorema Bayes. Teorema tersebut dikombinasikan dengan naive dimana diasumsikan kondisi antar atribut saling bebas [1]. Pada sebuah dataset, setiap baris/dokumen I diasumsikan sebagai vector dari nilai-nilai atribut x_1, x_2, x_3 ; dimana tiap nilai-nilai menjadi peninjauan atribut X_i ($i \in [1, n]$)).

Kelebihan Kekurangan Naive Bayes

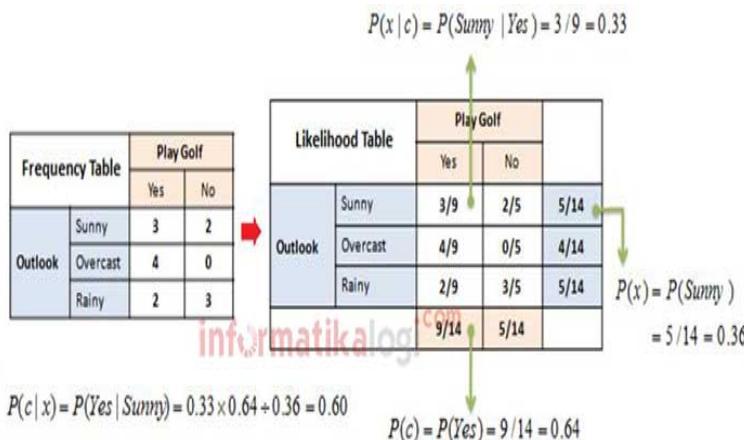
Kelebihan

- Mudah untuk dibuat
- Hasil bagus

Kekurangan

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

Persamaan Teorema Bayes



Gambar 4.17 Contoh Perhitungan Bayes

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

likelihood *Class Prior Probability*
Posterior Probability *Predictor Prior Probability*

Gambar 4.18 rumus bayes

Keterangan :

x : Data dengan class yang belum diketahui

c : Hipotesis data merupakan suatu class spesifik

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

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

$P(x|c)$: Probabilitas berdasarkan 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 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 4.19 rumus bayes

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 $(c|x_1, x_n)$ 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 4.20 rumus bayes

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 4.21 rumus bayes

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

Gambar 4.22 rumus bayes

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

$$P = (X_i = x_i \mid Y_i = y_i) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i - u_{ij})^2}{2\sigma^2}}$$

Gambar 4.23 Densitas Gauss

Keterangan :

P : Peluang

X_i : Atribut ke i

x_i : Nilai atribut ke i

Y : Kelas yang dicari

y_j : 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 4.24 Rumus Mean

Deviasi Standar

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

Gambar 4.25 Rumus Deviasi Standar

9. Classification and Regression Trees

CART (Classification And Regression Trees) yaitu metode pohon regresi dan pohon klasifikasi. Jika variabel dependen yang dimiliki bertipe kategorik maka *CART* menghasilkan pohon klasifikasi (*classification trees*), sedangkan jika variabel dependen yang dimiliki bertipe kontinu atau numerik maka *CART* menghasilkan pohon regresi (*regression trees*)

Terdapat tiga bagian penting dalam membuat pohon keputusan :

- Aturan pemisahan setiap simpul
- Aturan kriteria pemberhentian
- Penentuan nilai dugaan respon bagi setiap simpul akhir

Aturan pemisahan setiap simpul dibentuk melalui pemisahan gugus data dengan sederetan split biner sampai menghasilkan simpul akhir. Aturan pemisahannya adalah sebagai berikut :

1. Tiap pemisahan tergantung pada nilai yang berasal dari satu variabel penjelas.
2. Untuk variabel kontinu x_j yang diperbolehkan adalah $x_j \leq c$ dan $x_j > c$, dimana c adalah nilai tengah atau rata-rata antara dua nilai amatan pada peubah x_j secara berurutan. Jadi jika x_j mempunyai n nilai yang berbeda maka akan ada $n - 1$ pemisah (*midpoint*).
3. Untuk variabel kategorik, pemisahan yang terjadi berasal dari semua kemungkinan penyekatan berdasarkan terbentuknya dua anak gugus yang saling lepas (*disjoint*). Jika x_j variabel kategorik nominal dengan L kategori, maka akan ada $2^{L-1} - 1$ pemisah. Sedangkan jika berupa data kategorik ordinal maka akan ada $L-1$ pemisah.

Gambar 4.26 Aturan pemisahan

10. Adaboost

Model standard dari algoritma *adaboost* terdiri dari dua bagian, yaitu bagian *offline training* dan bagian *online recognizing*.

Bagian *offline training* adalah bagian proses pelatihan data yang tidak bekerja secara *realtime*. Bagian ini meliputi penginputan sampel gambar positif dan sampel gambar negatif, *preprocessing*, pelatihan data oleh algoritma *adaboost* sampai membangun detektor. Setelah detektor terbentuk kita bisa melakukan pendekripsi secara *realtime/online recognizing* terhadap data pengujian. Sebelum melakukan pendekripsi dengan algoritma *adaboost*, terlebih dahulu data pengujian sudah harus mengalami *preprocessing*.

4.1.4 Istilah dalam Data Mining

1. Knowledge Presentation

(di mana gambaran teknik visualisasi dan pengetahuan digunakan untuk memberikan pengetahuan yang telah ditambahkan kepada user).

2. Data Selection

(di mana data yang relevan dengan tugas analisis dikembalikan ke dalam database)

3. Data Transformation

(di mana data berubah atau bersatu menjadi bentuk yang tepat untuk menambah dengan ringkasan performa atau operasi agresi)

4. Data Cleaning

(untuk menghilangkan noise data yang tidak konsisten)

5. Data Integration

(di mana sumber data yang terpecah dapat disatukan)

6. Data Mining

(proses esensial di mana metode yang intelejen digunakan untuk mengekstrak pola data)

7. Pattern Evolution

(untuk mengidentifikasi pola yang benar-benar menarik yang mewakili pengetahuan berdasarkan atas beberapa tindakan yang menarik)

4.1.5 Fungsi Data Mining

Fungsi Data mining adalah mengidentifikasi fakta-fakta atau kesimpulan-kesimpulan yang di sarankan berdasarkan penyaringan melalui data untuk menjelajahi pola-pola atau anomali-anomali data.

Data Mining mempunyai 5 fungsi:

1. Classification, yaitu menyimpulkan definisi-definisi karakteristik sebuah grup.
Contoh: pelanggan-pelanggan perusahaan yang telah berpindah kesaingan perusahaan yang lain.
2. Clustering, yaitu mengidentifikasikan kelompok-kelompok dari barang-barang atau produk-produk yang mempunyai karakteristik khusus (clustering berbeda dengan classification, dimana pada clustering tidak terdapat definisi-definisi karakteristik awak yang di berikan pada waktu classification.)
3. Association, yaitu mengidentifikasikan hubungan antara kejadian-kejadian yang terjadi pada suatu waktu, seperti isi-isu dari keranjang belanja.

4. Sequencing Hampir sama dengan association, sequencing mengidentifikasi hubungan-hubungan yang berbeda pada suatu periode waktu tertentu, seperti pelanggan-pelanggan yang mengunjungi supermarket secara berulang-ulang.
5. Forecasting memperkirakan nilai pada masa yang akan datang berdasarkan pola-pola dengan sekumpulan data yang besar, seperti peramalan permintaan pasar.

4.1.6 Tujuan Data Mining

Tujuannya itu dapat dilihat dari dua sudut pandang, yaitu sudut pandang komersial dan sudut pandang keilmuan. Dari sudut pandang komersial, dataming 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:

- Explanatory adalah Untuk menjelaskan beberapa kondisi penelitian, seperti mengapa penjualan truk pick up meningkat di colorado.
- Confirmatory Untuk mempertegas hipotesis, seperti halnya 2 kali pendapatan keluarga lebih suka di pakai untuk membeli peralatan keluarga, di bandingkan dengan satu kali pendapatan keluarga.
- Exploratory Menganalisis data untuk hubungan yang baru yang tidak di harapkan, seperti halnya pola apa yang cocok untuk kasus penggelapan kartu kredit.
- mengetahui hilangnya pelanggan karena pesaing
- mengetahui item produk atau konsumen yang memiliki kesamaan karakteristik
- mengidentifikasi produk-produk yang terjual bersamaan dengan produk lain.
- memprediski tingkat penjualan
- menilai tingkat resiko dalam menentukan jumlah produksi suatu item.

- memprediksi prilaku bisnis di masa yang akan datang

4.1.7 Contoh Penerapan Data Mining

- Analisa pasar dan manajemen. Solusi yang dapat diselesaikan dengan data mining, diantaranya: Menembak target pasar, Melihat pola beli pemakai dari waktu ke waktu, Cross-Market analysis, Profil Customer, Identifikasi kebutuhan Customer, Menilai loyalitas Customer, Informasi Summary.
- Analisa Perusahaan dan Manajemen resiko. Solusi yang dapat diselesaikan dengan data mining, diantaranya: Perencanaan keuangan dan Evaluasi aset, Perencanaan sumber daya (Resource Planning), Persaingan (Competition).
- Sebuah perusahaan telekomunikasi menerapkan data mining untuk melihat dari jutaan transaksi yang masuk, transaksi mana sajakah yang masih harus ditan-gani secara manual.
- Financial Crimes Enforcement Network di Amerika Serikat baru-baru ini meng-gunakan data mining untuk me-nambang trilyunan dari berbagai subyek seperti property, rekening bank dan transaksi keuangan lainnya untuk mendeteksi transaksi-transaksi keuangan yang mencurigakan (seperti money laundry) .
- Australian Health Insurance Commision menggunakan data mining untuk meng-identifikasi layanan kesehatan yang sebenarnya tidak perlu tetapi tetap dilakukan oleh peserta asuransi.
- IBM Advanced Scout menggunakan data mining untuk menganalisis statistik permainan NBA (jumlah shots blocked, assists dan fouls) dalam rangka menca-pai keunggulan bersaing (competitive advantage) untuk tim New York Knicks dan Miami Heat.
- Jet Propulsion Laboratory (JPL) di Pasadena, California dan Palomar Obser-vatory berhasil menemukan 22 quasar dengan bantuan data mining. Hal ini merupakan salah satu kesuksesan penerapan data mining di bidang astronomi dan ilmu ruang angkasa.
- Internet Web surf-aid IBM Surf-Aid menggunakan algoritma data mining un-tuk mendata akses halaman Web khususnya yang berkaitan dengan pemasaran guna melihat prilaku dan minat customer serta melihat ke- efektif-an pemasaran

melalui Web.

4.1.8 Arsitektur Sistem Data mining

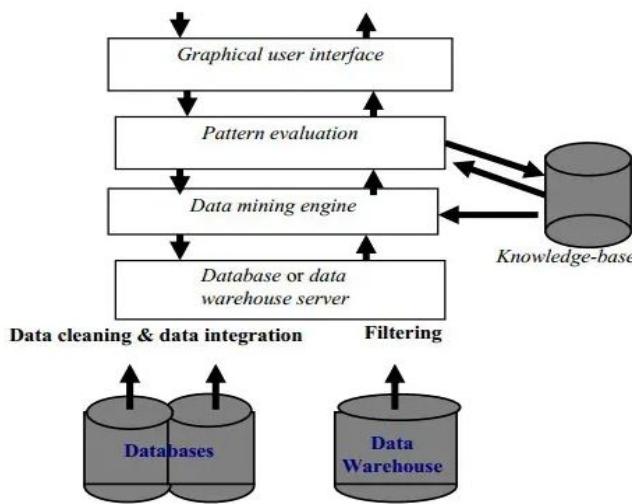
Data mining merupakan proses pencarian pengetahuan yang menarik dari data berukuran besar yang disimpan dalam basis data, data warehouse atau tempat penyimpanan informasi lainnya. Dengan demikian arsitektur sistem data mining memiliki komponen-komponen utama yaitu:

- Basis data, data warehouse atau tempat penyimpanan informasi lainnya.
- Basis data dan data warehouse Komponen ini bertanggung jawab dalam pengambilan relevant data, berdasarkan permintaan pengguna. Basis pengetahuan. Komponen ini merupakan domain knowledge yang digunakan untuk memandu pencarian atau mengevaluasi pola-pola yang dihasilkan. Pengetahuan tersebut meliputi hirarki konsep yang digunakan untuk mengorganisasikan atribut atau nilai atribut ke dalam level abstraksi yang berbeda.
- Pengetahuan tersebut juga dapat berupa kepercayaan pengguna (user belief), yang dapat digunakan untuk menentukan kemenarikan pola yang diperoleh. Contoh lain dari domain knowledge adalah threshold dan metadata yang menjelaskan data dari berbagai sumber yang heterogen.
- Data mining. Bagian ini merupakan komponen penting dalam arsitektur sistem data mining. Komponen ini terdiri modul-modul fungsional data mining seperti karakterisasi, asosiasi, klasifikasi, dan analisis cluster.
- Modul evaluasi pola. Komponen ini menggunakan ukuran-ukuran kemenarikan dan berinteraksi dengan modul data mining dalam pencarian pola-pola menarik. Modul evaluasi pola dapat menggunakan threshold kemenaraikan untuk memfilter pola-pola yang diperoleh.
- Antarmuka pengguna grafis. Modul ini berkomunikasi dengan pengguna dan sistem data mining. Melalui modul ini, pengguna berinteraksi dengan sistem dengan menentukan kueri atau task data mining.

Antarmuka juga menyediakan informasi untuk memfokuskan pencarian dan melakukan eksplorasi data mining berdasarkan hasil data mining antara.

- Komponen ini juga memungkinkan pengguna untuk mencari (browse) basis data dan skema data warehouse atau struktur data, evaluasi pola yang diperoleh dan visualisasi pola dalam berbagai bentuk.

Arsitektur sebuah sistem data mining dapat dilihat dibawah ini



Gambar 4.27 Evaluasi Pola Data Mining

Data mining dapat diaplikasikan pada berbagai jenis penyimpanan data seperti basis data relational, data warehouse, transactional database, object-oriented and object-relational databases, spatial databases, time-series data and temporal data, text databases and multimedia databases, heterogeneous and legacy databases dan WWW.

1. Basis data Relasional

Basis data relasional merupakan koleksi dari table. Setiap table berisi atribut (field) dan biasanya menyimpan sejumlah besar tuple (record).

Setiap tuple dalam table relasional merepresentasikan sebuah objek yang diidentifikasi oleh kunci unik dan dideskripsikan oleh sekumpulan nilai atribut. Data relasional dapat diakses oleh kueri basis data yang ditulis dalam bahasa kueri relasional seperti SQL atau dengan bantuan antarmuka pengguna grafis.

2. Data warehouse

Data warehouse merupakan tempat penyimpanan informasi yang dikumpulkan dari berbagai sumber, disimpan dalam skema yang dipersatukan (unified schema) dan biasanya bertempat pada tempat penyimpanan tunggal.

Data warehouse dikonstruksi melalui sebuah proses data cleaning, data transformation, data integration, data loading dan periodic data refreshing. Untuk memfasilitasi proses pembuatan keputusan, data dalam data warehouse diorganisasikan ke dalam subjek utama seperti customer, item, supplier atau aktivitas.

Data disimpan untuk menyediakan informasi dari perspektif sejarah (seperti 5-10 tahun yang lalu) dan biasanya data tersebut diringkas (summarized). Sebagai contoh, daripada menyimpan data rinci dari transaksi penjualan, data warehouse dapat menyimpan ringkasan dari transaksi per tipe item untuk setiap toko atau diringkas dalam level yang lebih tinggi seperti daerah pemasaran.

Data warehouse biasanya dimodelkan oleh struktur basis data multidimensional, dimana setiap dimensi berkaitan dengan sebuah atribut atau sekumpulan atribut dalam skema, dan setiap sel menyimpan nilai dari ukuran agregasi seperti count dan sales amount.

Struktur fisik dari data warehouse dapat berupa penyimpanan basis data relasional atau sebuah kubus data multidimensional.

Selain data warehouse, terdapat istilah penyimpanan data yang lain yaitu data mart. Sebuah data warehouse mengumpulkan informasi mengenai subjek-subjek yang menjangkau seluruh organisasi, dengan demikian cakupannya enterprise-wide.

Sedangkan data mart merupakan sub bagian dari data warehouse. Fokus data mart adalah pada subjek yang dipilih dan dengan demikian cakupannya adalah department-wide.

3. Basis data Transaksional

Secara umum, basis data transaksional terdiri dari sebuah file dimana setiap record merepresentasikan transaksi. Sebuah transaksi biasanya meliputi bilangan identitas transaksi yang unik (trans id), dan sebuah daftar dari item yang membuat transaksi (seperti item yang dibeli dalam sebuah took).

Basis data transaksi dapat memiliki tabel tambahan, yang mengandung informasi lain berkaitan dengan penjualan seperti tanggal transaksi, customer ID number, ID number dari sales person dan dari kantor cabang (branch) dimana penjualan terjadi.

4.1.9 Software Aplikasi Data Mining

Daftar software dan aplikasi bersifat free dan open source yang dapat digunakan untuk data mining antara lain sebagai berikut:

1. Carrot2

framework clustering teks dan hasil pencarian.

2. org

miner struktur kimia dan web search engine.

3. ELKI

project set universitas dengan analisis cluster canggih dan metode deteksi outlier yang ditulis dalam bahasa Java.

4. GATE

pemroses bahasa natural dan tool rekayasa bahasa.

5. JHepWork

framework analisis data berbasis Java yang dikembangkan di Argonne National Laboratory.

6. KNIME

Konstanz Information Miner, framework analisis data komprehensif yang user-friendly.

Perangkat lunak ini sangat cocok digunakan bagi pemula karena KNIME tidak perlu memasukkan query secara manual yang biasanya harus dilakukan perangkat lunak lain seperti R. Pada perangkat lunak KNIME ini system builder hanya perlu memilih node-node dan menyambungkan antar node tersebut layaknya

sebuah flowchart.

Setiap node merepresentasikan aktivitas atau logika pada pemrograman. Sebagai contoh, apabila aktivitas yang harus dilakukan oleh program tersebut adalah membaca data maka system builder hanya perlu memasukkan node file reader dan selanjutnya mengkonfigurasikan data yang hendak dibaca. Selain memudahkan, KNIME pun dapat dikonfigurasikan dengan R atau python.

7. ML-Flex

paket software yang memungkinkan pengguna untuk mengintegrasikan dengan paket mesin belajar pihak ketiga yang ditulis dalam bahasa pemrograman apapun, mengeksekusi klasifikasi analisis secara paralel sepanjang node komputasi, dan menghasilkan laporanHTML dari hasil klasifikasi.

8. NLTK (Natural Language Toolkit)

Sebuah kumpulan libraries dan program untuk pengolahan bahasa simbolik dan statistik alami untuk bahasa Python.

NLTK adalah salah satu tool yang sangat populer pada ilmu Natural Language Processing (NLP) dengan menggunakan bahasa pemrograman Python. Natural Language Toolkit ini sangat mendukung proses pengolahan bahasa natural seperti classification, tokenization, stemming, tagging, parsing dll.

- Instalasi

Disini telah diasumsikan telah memasang interpreter Python di sistem operasi. Kemudian dilanjutkan dengan proses instalasi NLTK dengan menggunakan pip

```
1 pip install -U nltk
```

Gambar 4.28 Proses Instalasi

Selanjutnya kita diharuskan mengunduh paket tambahan dari modul NLTK ini dengan menjalankan kode dibawah.

```
1 import nltk  
2 nltk.download()
```

Gambar 4.29 Proses Import

Akan muncul jendela manajemen paket NLTK, pilih paket yang ingin diinstall. Jika tidak yakin dengan paket-paket apa saja yang harus diinstall kita bisa menginstall semua paket yang tersedia.

The screenshot shows a software window titled 'NLTK Data Downloader'. At the top, there's a menu bar with 'File', 'View', 'Sort', and 'Help'. Below the menu is a tab bar with four tabs: 'Collections' (selected), 'Corpora', 'Models', and 'All Packages'. A main table lists various NLTK data packages with columns for Identifier, Name, Size, and Status. The table includes rows for 'all', 'all-corpora', 'all-nltk', 'book', 'popular', 'tests', and 'third-party'. The 'Status' column indicates all packages are 'not installed'. At the bottom of the window are two buttons: 'Download' and 'Refresh'. Below the buttons, there are two text input fields: 'Server Index: https://raw.githubusercontent.com/nltk/nltk_data/gh-pages/packages.html' and 'Download Directory: /home/devtrik/nltk_data'.

Identifier	Name	Size	Status
all	All packages	n/a	not installed
all-corpora	All the corpora	n/a	not installed
all-nltk	All packages available on nltk_data gh-pages branch	n/a	not installed
book	Everything used in the NLTK Book	n/a	not installed
popular	Popular packages	n/a	not installed
tests	Packages for running tests	n/a	not installed
third-party	Third-party data packages	n/a	not installed

Gambar 4.30 Proses Pemilihan Paket Install

Setelah berhasil menginstal paket-paket yang dibutuhkan, NLTK sudah siap kita gunakan dalam proses Natural Language Processing.

9. Orange

Sebuah komponen berbasis data mining dan suitesoftware mesin pembelajaran ditulis dalam bahasa Python.

10. R

Sebuah bahasa pemrograman dan lingkungan perangkat lunak untuk komputasi statistik, data mining, dan grafis. Ini adalah bagian dari proyek GNU.

11. RapidMiner

Sebuah lingkungan untuk pembelajaran mesin dan eksperimen data mining.

12. UIMA

framework komponen untuk menganalisis konten tidak terstruktur seperti teks, audio dan video-awalnya dikembangkan oleh IBM

13. Weka

suitesoftware aplikasi pembelajaran mesin yang ditulis dalam bahasa pemrograman Java.

4.2 *Clustering*

4.2.1 Definisi

Clustering Clustering adalah suatu metode pengelompokan berdasarkan ukuran kedekatan atau (kemiripan). Clustering beda dengan group, kalau group berarti kelompok yang sama, kondisinya kalau tidak pasti bukan kelompoknya. Namun kalau cluster tidak harus sama hanya kelompok berdasar yang kedekatan dari suatu karakteristik sample yang ada, yaitu dengan menggunakan rumus jarak eucledian distance. Aplikasinya cluster sangat banyak, karena hampir dalam mengidentifikasi permasalahan atau pengambilan keputusan tidak sama persis namun tetapi cenderung memiliki kemiripan saja.

Clustering juga disebut sebagai segmentasi data dalam beberapa aplikasi. Clustering bersifat unsupervised learning karena tidak membutuhkan proses training dalam pengelompokan kelasnya, tidak seperti klasifikasi yang bersifat supervised learning. Karena itu, clustering yang bersifat learning by observation lebih baik dari klasifikasi yang bersifat learning by example. Dalam data mining difokuskan untuk menemukan metode yang lebih efisien dan efektif untuk analisis cluster dalam database ukuran besar

4.2.2 Macam- macam *Clustering*

Metode Clustering dasarnya mengoptimalkan pusat cluster (centroid) atau mengoptimalkan lebar antara cluster. Secara umum metode Clustering dibagi menjadi dua metode, yaitu:

1. Metode *Hierarki*

Metode ini digunakan untuk mencari struktur pengelompokan dari obyek-obyek. Jadi, hasil pengelompokannya disajikan secara hierarki atau berjenjang. Metode hierarki ini terdiri dari dua cara, yaitu : *Agglomerative Clustering* (penggabungan) dan *Divisive Clustering* (pemecahan).

2. Metode *Non Hierarki*

Metode ini digunakan apabila jumlah kelompok yang diinginkan diketahui dan biasanya dipakai untuk mengelompokkan data yang ukurannya besar. Metode yang digunakan adalah K-Means Clustering.

4.2.3 Pendekatan Clustering

4.2.3.1 Clustering Dengan Pendekatan Partisi

1. Kmeans

Salah satu metode yang banyak digunakan dalam melakukan clustering dengan partisi ini adalah metode k-means.

2. Mixture Modelling (Mixture Modeling)

Mixture modelling (mixture modeling) merupakan metode pengelompokan data yang mirip dengan k-means dengan kelebihan penggunaan distribusi statistik dalam mendefinisikan setiap cluster yang ditemukan. Dibandingkan dengan k-means yang hanya menggunakan cluster center, penggunaan distribusi statistik ini mengijinkan kita untuk:

- a. Memodel data yang kita miliki dengan setting karakteristik yang berbeda-beda
- b. Jumlah cluster yang sesuai dengan keadaan data bisa ditemukan seiring dengan proses pemodelan karakteristik dari masing-masing cluster
- c. Hasil pemodelan clustering yang dilaksanakan bisa diuji tingkat keakuratannya

Distribusi statistik yang digunakan bisa bermacam-macam mulai dari yang digunakan untuk data categorical sampai yang continuous, termasuk di antaranya distribusi binomial, multinomial, normal dan lain-lain. Beberapa distribusi yang bersifat tidak normal seperti distribusi Poisson, von-Mises, Gamma dan Student t, juga diterapkan untuk bisa mengakomodasi berbagai keadaan data yang ada di lapangan. Beberapa pendekatan multivariate juga banyak diterapkan untuk memperhitungkan tingkat keterkaitan antara variabel data yang satu dengan yang lainnya.

4.2.3.2 Clustering dengan Pendekatan Hirarki Clustering dengan pendekatan hirarki mengelompokkan data yang mirip dalam hirarki yang sama dan yang tidak mirip di hirarki yang agak jauh. Ada dua metode yang sering diterapkan yaitu agglomerative hierarchical clustering dan divisive hierarchical clustering. Agglomerative melakukan proses clustering dari N cluster menjadi satu kesatuan cluster, dimana N adalah jumlah data, sedangkan divisive melakukan proses clustering yang sebaliknya yaitu dari satu cluster menjadi N cluster.

Beberapa metode hierarchical clustering yang sering digunakan dibedakan menu-
rut cara mereka untuk menghitung tingkat kemiripan. Ada yang menggunakan Sin-
gle Linkage, Complete Linkage, Average Linkage, Average Group Linkage dan
lain-lainnya. Seperti juga halnya dengan partition-based clustering, kita juga bisa
memilih jenis jarak yang digunakan untuk menghitung tingkat kemiripan antar data.

Salah satu cara untuk mempermudah pengembangan dendrogram untuk hierarchi-
cal clustering ini adalah dengan membuat similarity matrix yang memuat tingkat
kemiripan antar data yang dikelompokkan. Tingkat kemiripan bisa dihitung dengan
berbagai macam cara seperti dengan Euclidean Distance Space. Berangkat dari sim-
ilarity matrix ini, kita bisa memilih lingkage jenis mana yang akan digunakan untuk
mengelompokkan data yang dianalisa.

4.2.3.3 Clustering Dengan Pendekatan Automatic Mapping

1. Self-Organising Map (SOM)

Self-Organising Map (SOM) merupakan suatu tipe Artificial Neural Networks
yang di-training secara unsupervised. SOM menghasilkan map yang terdiri dari
output dalam dimensi yang rendah (2 atau 3 dimensi). Map ini berusaha men-
cari property dari input data. Komposisi input dan output dalam SOM mirip
dengan komposisi dari proses feature scaling (multidimensional scaling).

Walaupun proses learning yang dilakukan mirip dengan Artificial Neural Net-
works, tetapi proses untuk meng-assign input data ke map, lebih mirip dengan
K-Means dan kNN Algorithm.

4.2.4 Hal-hal Terkait Dengan Clustering

- Analisa Faktor
- Singular Value Decomposition
- Eigen Value and Eigen Vector
- Similarity Measure
- Feature Discretisation
- Feature Selection
- Feature Scaling
- Indexing Method For Searching

4.2.5 Tahap validasi Cluster

Melakukan validasi dan profiling cluster. Cluster yang terbentuk kemudian diuji apakah hasil tersebut valid. Kemudian dilakukan proses profiling untuk menjelaskan karakteristik setiap cluster berdasarkan profil tertentu (seperti usia konsumen pembeli rumah, tingkat penghasilannya dan sebagainya).

Analisis cluster agak bersifat subjektif dalam penentuan penyelesaian cluster yang optimal, sehingga peneliti seharusnya memberikan perhatian yang besar mengenai validasi dan jaminan tingkat signifikansi pada penyelesaian akhir dari cluster. Meskipun tidak ada metode untuk menjamin validitas dan tingkat signifikansi , beberapa pendekatan telah dikemukakan untuk memberikan dasar bagi perkiraan peneliti.

- Validasi Hasil Cluster

Validasi termasuk usaha yang dilakukan oleh peneliti untuk menjamin bahwa hasil cluster adalah representatif terhadap populasi secara umum, dan dengan demikian dapat digeneralisasi untuk objek yang lain dan stabil untuk waktu tertentu.

Pendekatan langsung dalam hal ini adalah dengan analisis sample secara terpisah kemudian membandingkan antara hasil cluster dengan perkiraan masing-masing cluster. Pendekatan ini sering tidak praktis, karena adanya keterbatasan waktu dan biaya atau ketidaktersediaan objek untuk perkalian analisis cluster. Dalam hal ini pendekatan yang biasa digunakan adalah dengan membagi sample menjadi dua kelompok. Masing-masing dianalisis cluster secara terpisah, kemudian hasilnya dibandingkan.

- Profiling Hasil Cluster

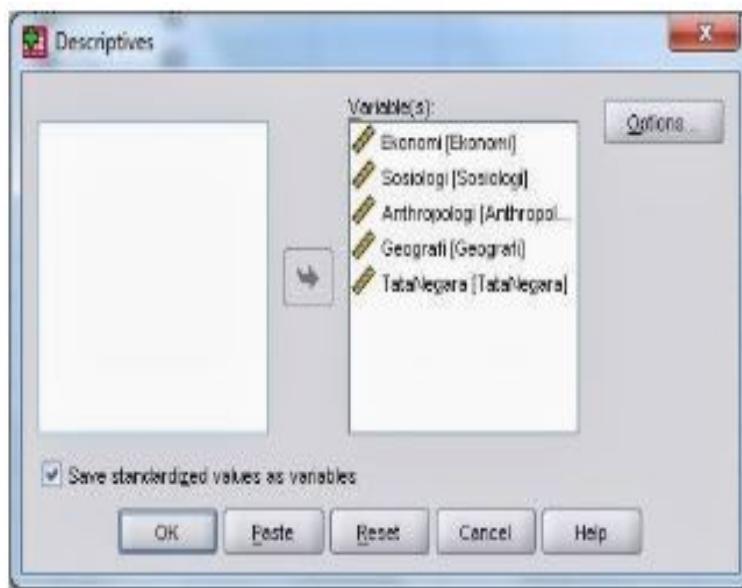
Tahap Profiling meliputi penggambaran karakteristik masing-masing cluster untuk menjelaskan bagaimana mereka bisa berbeda secara relevan pada tiap dimensi.

Tipe ini meliputi penggunaan analisis diskriminan. Prosedur dimulai setelah cluster ditentukan. Peneliti menggunakan data yang sebelumnya tidak masuk dalam prosedur cluster untuk menggambarkan karakteristik masing-masing cluster. Meskipun secara teori tidak masuk akal (rasional) dalam perbedaan silang cluster, akan tetapi hal ini diperlukan untuk memprediksi validasi tak-siran, sehingga minimal penting secara praktik.

4.2.5.1 Analisis Cluster Hirarki dengan SPSS

Langkah-langkah Cluster Hirarki menggunakan SPSS

Pada menu SPSS, klik Analyze, Descriptives Statistics, Descriptives. Setelah jendela terbuka, maka langkah selanjutnya adalah memasukkan semua variabel ke kotak Variable(s) dan jangan lupa centang Save standardized values as variables yang artinya kita memerintahkan SPSS untuk otomatis menyimpan nilai Z Score atau nilai variabel yang telah distandardisasi.



Gambar 4.31
Cluster Hirarki Dengan SPSS

Setelah anda klik OK, maka lihat pada dataset bahwa telah muncul variabel z-score tiap variabel asli. Dasar dari perhitungan nilai z-score adalah hasil uji descriptive tadi yang dapat dilihat pada output view sebagai berikut:

Descriptive Statistics					
	N	Minimum	Maximum	Mean	Std. Deviation
Ekonomi	14	56	89	76,64	9,779
Sosiologi	14	61	95	74,64	10,278
Anthropologi	14	50	98	69,07	16,050
Geografi	14	52	90	69,43	13,910
TataNegara	14	60	90	74,00	10,599
Valid N (listwise)	14				

Gambar 4.32 Output Deskriptive Statistics Analisis Cluster SPSS

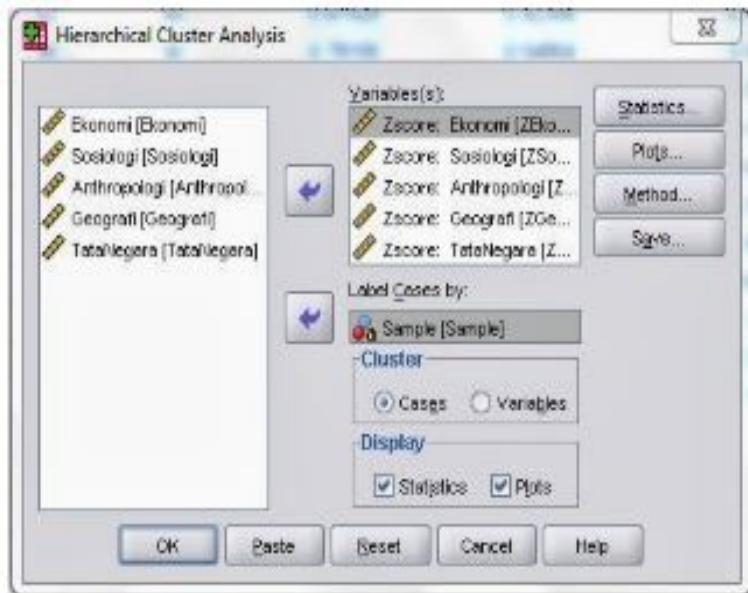
Setelah kita mendapatkan variabel z-score, maka perlu dipahami bahwasanya variabel z-score inilah yang akan kita gunakan dalam analisis cluster selanjutnya.

Konsep Analisis Cluster Hirarki

Konsep dari metode hirarkis ini dimulai dengan menggabungkan 2 obyek yang paling mirip, kemudian gabungan 2 obyek tersebut akan bergabung lagi dengan satu atau lebih obyek yang paling mirip lainnya. Proses clustering ini pada akhirnya akan menggumpal menjadi satu cluster besar yang mencakup semua obyek. Metode ini disebut juga sebagai metode aglomerativ yang digambarkan dengan dendrogram.

Langkah Analisis Cluster

Langkah berikutnya pada menu, klik Analyze, Classify, Hierarchical Cluster. Kemudian masukkan variabel-variabel z-score ke kotak Variable(s) dan variabel Sample ke kotak Label Cases By. Pada cluster pilih Cases dan pada Display pilih Statistics dan Plots.



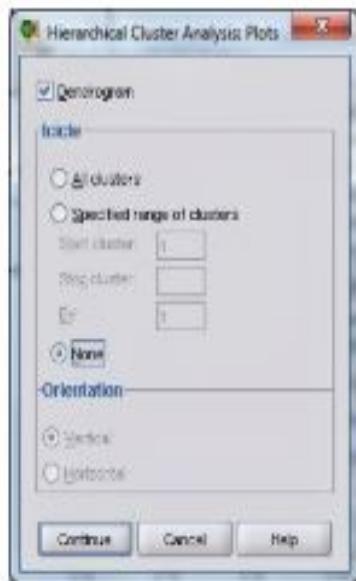
Gambar 4.33 Analisis Cluster Hirarki dengan SPSS

Klik tombol Statistics, centang kotak Agglomeration Schedule dan Proximity Matrix. Proximity matrix berguna untuk menampilkan jarak antar variabel. Pada Cluster membership, klik Range of Solution lalu ketik 2 pada From atau Minimum number of cluster dan 4 pada Trough atau Maximum number of cluster (maksudnya adalah kita akan menampilkan susunan 2,3 dan 4 cluster).



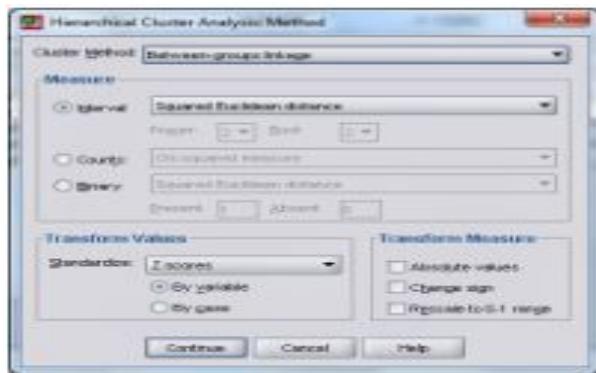
Gambar 4.34 Agglomeration Schedule

Setelah klik Continue, maka anda klik tombol Plots kemudian centang Dendogram dan pada Icicle pilih None.



Gambar 4.35 Dendogram

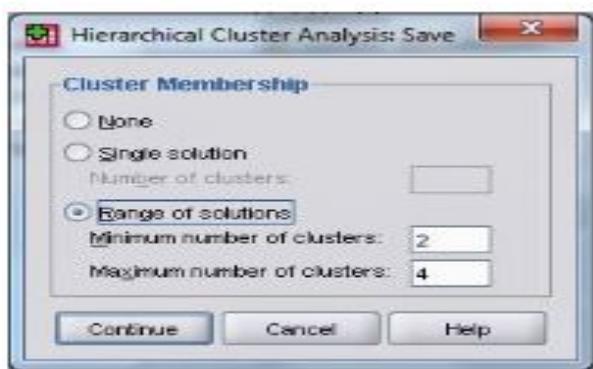
Setelah klik Continue, maka anda klik tombol Method. Setelah muncul jendela, pada Cluster Method pilih Between-group linkage. Pada measure pilih interval kemudian pilih Squared Euclidian Distance. Kemudian pada Transform Values pilih Z-Score. Kemudian klik Continue.



Gambar 4.36 Method Cluster

Klik tombol Save kemudian plih range of solutions dan pada minimum number of cluster isi dengan nilai 2. Sedangkan pada minimum of cluster isi dengan

nilai 4.



Gambar 4.37 Membership Cluster

4.2.6 K-Means Clustering

Algoritma *K-Means Clustering* merupakan salah satu algoritma dengan partitional, karena *K-Means Clustering* didasarkan pada penentuan jumlah awal kelompok dengan mendefinisikan nilai centroid awalnya. Dibutuhkan jumlah cluster awal yang diinginkan sebagai masukan dan menghasilkan titik centroid akhir sebagai output. Metode *K-Means Clustering* akan memilih pola k sebagai titik awal centroid secara acak atau random. Jumlah iterasi untuk mencapai *cluster centroid* akan dipengaruhi oleh calon *cluster centroid* awal secara *random*. Sehingga didapat cara dalam pengembangan algoritma dengan menentukan *centroid cluster* yang dilihat dari kepadatan data awal yang tinggi agar mendapatkan kinerja yang lebih tinggi.

Dalam penyelesaiannya, algoritma *K-Means Clustering* akan menghasilkan titik *centroid* yang dijadikan tujuan dari algoritma *K-Means Clustering*. Setelah iterasi *K-Means Clustering* berhenti, setiap objek dalam dataset menjadi anggota dari suatu *cluster*. Nilai *cluster* ditentukan dengan mencari seluruh objek untuk menemukan cluster dengan jarak terdekat ke objek. Algoritma *K-Means Clustering* akan mengelompokan item data dalam suatu dataset ke suatu *cluster* berdasarkan jarak terdekat. Nilai *centroid* awal yang dipilih secara acak yang menjadi titik pusat awal, akan dihitung jarak dengan semua data menggunakan rumus *Euclidean Distance*. Data yang memiliki jarak pendek terhadap *centroid* akan membuat sebuah *cluster*. Proses ini berkelanjutan sampai tidak terjadi perubahan pada setiap kelompok.

4.2.7 Pengelompokan untuk Pemahaman (*Understanding*)

Pengelompokan untuk pemahaman bertujuan menghasilkan kelompok-kelompok yang terdiri dari objek-objek dengan karakteristik yang serupa, seperti halnya manusia mengelompokkan objek-objek.

1. Aplikasi di Bidang Biologi

K-means dapat digunakan untuk mengelompokkan gen berdasarkan polanya. Hal ini diperlukan untuk menemukan gen yang memiliki fungsi serupa.

2. Aplikasi di Bidang Bisnis

K-means dapat digunakan untuk melakukan segementasi pasar. Segementasi pasar adalah pengelompokan pelanggan sesuai karakteristik mereka (misalnya: gaya hidup, kebutuhan). *K-means* juga dapat digunakan dalam sistem pemberi rekomendasi untuk mengelompokkan objek-objek yang saling terkait.

3. Aplikasi di Bidang Temu Kembali Informasi

K-means dapat digunakan untuk mengelompokkan dokumen sehingga memudahkan temu kembali dokumen berdasarkan topiknya.

4.2.8 Pengelompokan untuk *Utility*

Pengelompokan bertujuan untuk mengelompokkan himpunan data yang besar untuk memudahkan analisis data atau pemrosesan data lebih lanjut. Untuk tujuan ini, centroid dari cluster memegang peran lebih berarti.

1. Kompresi Data Multimedia

K-means dapat digunakan untuk kompresi data multimedia (citra, audio, video). Setiap objek dalam data (misalnya *pixel* dari citra) direpresentasikan dengan *centroid* dari *cluster* yang memuat objek tersebut. Teknik kompresi ini disebut juga kuantisasi vektor.

2. Rangkuman Data

K-means dapat digunakan untuk mengelompokkan data sebelum menerapkan teknik analisis data lainnya seperti regresi, tetangga terdekat, atau PCA. *K-means* dapat digunakan untuk terlebih dahulu mengelompokkan data ke dalam *cluster-cluster*. Kemudian teknik analisis data hanya perlu diterapkan pada *centroid* dari setiap *cluster* sehingga lebih efisien dalam hal penggunaan waktu dan ruang.

4.2.9 Variasi

Terdapat beberapa algoritma yang merupakan pengembangan/variasi dari algoritma *k-means*

1. K-means++

Algoritma untuk memilih nilai awal untuk algoritma k-means[6]. Algoritma ini digunakan untuk mengurangi dampak buruk algoritma k-means yang sangat tergantung dari nilai awalnya.

2. K-medoids

Algoritma clustering yang berbasiskan prototype/model dari cluster. K-means menggunakan centroid (rata-rata) sebagai model dari cluster, sedangkan K-medoids menggunakan medoid (median).

3. Bisecting K-means

Ide dasarnya adalah menggunakan K-means untuk membagi dua suatu cluster. Awalnya setiap objek tergabung dalam satu cluster. Pada tiap iterasi, pilih satu cluster untuk dibagi dua menggunakan K-means. Hal ini dilakukan hingga terbentuk K cluster. Algoritma bisecting K-means bekerja lebih cepat dari K-means karena mengurangi jumlah objek yang diperbandingkan pada setiap iterasi.

Berikut ini langkah-langkah yang terdapat pada algoritma K-Means Clustering:

1. Tentukan k sebagai jumlah *cluster* yang dibentuk Untuk menentukan banyaknya *cluster* k dilakukan dengan beberapa pertimbangan seperti pertimbangan teoritis dan konseptual yang mungkin diusulkan untuk menentukan berapa banyak *cluster*.
2. Tentukan k *centroid* (titik pusat *cluster*) awal secara random Menentukan *centroid* awal dilakukan secara *random* atau acak dari objek-objek yang disediakan sebanyak k *cluster*, selanjutnya untuk menghitung *centroid* *cluster* ke- i berikutnya
3. Hitung jarak setiap objek ke masing-masing *centroid* dari masing-masing *cluster*.
4. Alokasikan masing-masing objek ke dalam *centroid* yang paling dekat. Untuk melakukan pengalokasian objek kedalam masing masing *cluster* pada saat iterasi secara umum dapat dilakukan dengan cara *hard k-means* dimana secara tegas setiap objek dinyatakan sebagai anggota *cluster* dengan mengukur jarak kedekatan sifatnya terhadap titik pusat *cluster* tersebut.

5. Lakukan iterasi, kemudian tentukan posisi *centroid* baru dengan menggunakan persamaan .
6. Ulangi langkah 3 jika posisi *centroid* baru tidak sama.

4.2.9.1 Langkah-langkah Clustering Menggunakan R Salah satu metode yang banyak digunakan dalam melakukan clustering dengan partisi atau non-hierarchical clustering ini adalah metode k-means.

Metode K-Means Clustering bertujuan untuk meminimalisasikan objective function yang diset dalam proses clustering dengan cara meminimalkan variasi antar data yang ada di dalam suatu cluster dan memaksimalkan variasi dengan data yang ada di cluster lainnya.

Contoh: Data yang digunakan adalah seperti berikut:

No	Satker	Tahanan Dewasa	Tahanan Anak	Napi Dewasa	Napi Anak	No	Satker	Tahanan Dewasa	Tahanan Anak	Napi Dewasa	Napi Anak
1	ACEH	1828	15	4425	62	23	PAPUA	368	6	1123	29
2	BALI	742	8	1537	9	24	PAPUA BI	202	1	626	12
3	BANGKAI	512	9	1349	21	25	RIAU	2524	35	7323	115
4	BANTEN	2376	18	5209	83	26	SULAWESI	264	3	401	11
5	BENGKULU	456	10	1498	48	27	SULAWESI	3346	126	4660	139
6	D.I.YOGY	461	12	928	15	28	SULAWESI	796	7	1438	26
7	DKI JAKAI	6991	35	8774	24	29	SULAWESI	767	23	1238	29
8	GORONTI	192	2	614	8	30	SULAWESI	800	38	1464	29
9	JAMBI	1035	12	2879	65	31	SUMATER	1236	13	3057	37
10	JAWA BAR	5135	79	15698	235	32	SUMATER	2826	64	7589	237
11	JAWA TEN	3031	25	7349	156	33	SUMATER	9467	86	15874	154
12	JAWA TIN	7051	124	12549	170						
13	KALIMAN	1327	9	2607	47						
14	KALIMAN	2032	24	6218	56						
15	KALIMAN	1063	10	2420	24						
16	KALIMAN	3086	30	5816	88						
17	KEPULAU	1032	5	2899	56						
18	LAMPUNG	2392	59	3961	210						
19	MALUKU	307	7	705	20						
20	MALUKU	274	1	680	14						
21	NUSA TEN	759	12	1410	45						
22	NUSA TEN	566	2	2427	37						

Gambar 4.38 Contoh Data yang Digunakan

Dari data tersebut dilakukan klasterisasi menggunakan R dengan langkah-langkah sebagai berikut

1. Import data yang digunakan ke dalam aplikasi R.

```
data=read.csv("D:\\PENTING!!!\\kuliah semester 6\\prak SMT\\
pertemuan 8\\lpaoran 7 SMT\\Data Laporan SMT BAB
7 analisis non hirarki metode k-means.csv")
```

```
data
```

Gambar 4.39 Contoh Input Data

2. Kemudian membuat data baru dimana pada data tersebut hanya terdapat variabel yang numeric dan kemudian menghitung statistika deskriptif dari data

```
data_nina<-data[,c(-1,-2)]
data_nina
summary(data_nina)
```

Gambar 4.40 Memilih Variabel Numerik dan Menghitung Statistik Deskriptif

didapatkan hasil perhitungan statistika deskriptif yaitu seperti berikut

> summary(data_nina)			
Tahanan.Dewasa	Tahanan.Anak	Napi.Dewasa	Napi.Anak
Min. : 192	Min. : 1.00	Min. : 401	Min. : 8.00
1st Qu.: 512	1st Qu.: 7.00	1st Qu.: 1349	1st Qu.: 24.00
Median :1035	Median :12.00	Median :2607	Median :45.00
Mean :1978	Mean :27.58	Mean :4144	Mean :70.03
3rd Qu.:2524	3rd Qu.:35.00	3rd Qu.:5816	3rd Qu.:88.00
Max. :9467	Max. :126.00	Max. :15874	Max. :237.00

Gambar 4.41 Hasil Statistika Deskriptif

Keterangan:

Berdasarkan statistika deskriptif pada gambar 3.1 diatas praktikan akan memberikan ilustrasi penjelasan untuk variabel Tahanan Dewasa, dimana didapatkan bahwa untuk variabel Tahanan dewasa didapatkan nilai minimum atau jumlah paling sedikit adalah 192, dan jumlah tahanan dewasa paling banyak adalah 9467 dengan rata-rata jumlah tahanan dewasa sebesar 192 , nilai tengah atau median sebesar 1035, nilai kuartil-1 sebesar 512 dan nilai kuartil-3 sebesar 2524 serta dapat dilihat bahwa pada variabel tahanan dewasa tidak terdapat data missing.

3. Menghitung jarak dari masing-masing data ke data lainnya dengan syntax:

```
#corelation distance
library(factoextra)
distance <- get_dist(data_nina) ##jarak antara obyek yang satu dengan
distance
fviz_dist(distance, gradient=list(low="green",mid="white",high="red"))
```

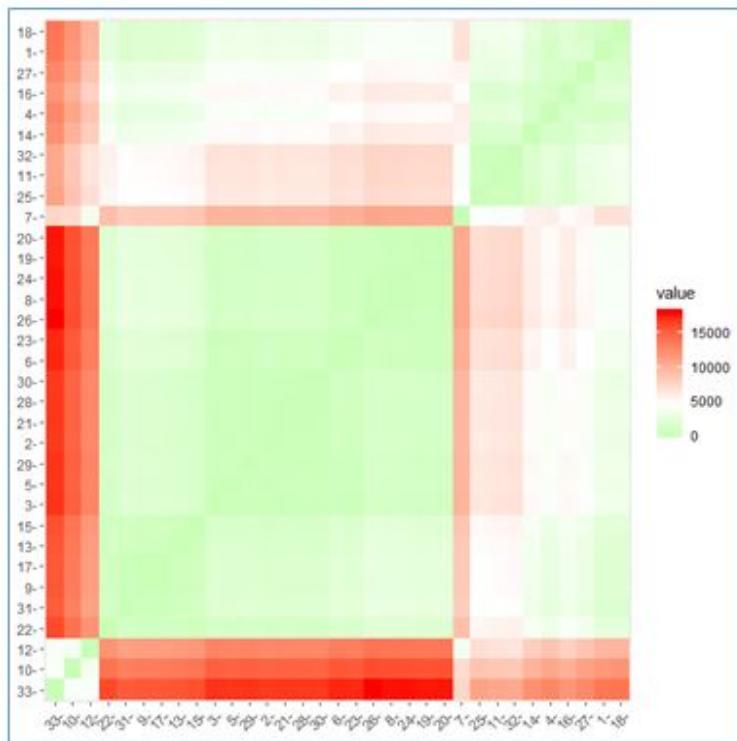
Gambar 4.42 Menghitung Jarak Antar Objek

didapatkan hasil:

> distance	1	2	3	4	5	6	7	8	9	10	11	12
2	3085.90311											
3	3345.94516	297.30288										
4	956.77061	4019.84030	4286.35941									
5	3215.86287	252.11505	152.27278	4160.19351								
6	3754.98549	670.74138	424.13088	4690.29274	572.02972							
7	6750.71952	9561.64860	854.37725	5831.91392	9753.27484	1007.90008						
8	4147.66834	1074.46080	801.77491	5088.19831	935.70081	413.64961	10621.36272					
9	1737.52209	1374.75998	1617.12098	2688.40864	1482.55691	2034.30013	8380.16533	247.48278				
10	11749.50224	14828.63740	10077.0311	10487.02849	14939.88122	13493.61368	7171.67626	15875.06734	13459.94934			
11	3163.21371	6248.26080	6508.75119	2239.19695	6377.48501	6917.67237	4210.67085	7310.44315	4896.26041	8610.56877		
12	9659.32969	12692.79331	12970.5022	8703.44932	12849.92241	13360.84990	3779.34677	13767.03432	11389.67625	3686.93735	6573.46157	
13	1885.83828	1220.06967	1499.15476	2805.74090	1385.80085	1889.45230	8373.40970	2393.87096	399.47591	13635.07935	5040.07113	13473.26867
14	1804.60023	4855.75185	5100.88315	1066.38736	4965.66357	5158.51121	5579.06462	5898.57844	3484.70300	9976.76755	1512.31691	8080.52709
15	2146.32686	939.65898	1204.43016	3083.18585	1082.65830	1608.88007	8689.94390	2005.14264	461.68171	13890.13139	5309.02194	11768.95409
16	1875.72546	4879.64159	5156.05142	934.19377	5035.39710	5548.76761	4899.27648	5933.42120	3582.37673	10093.38075	1535.50090	781.73442
17	1721.17053	1393.32767	1635.28010	2672.70163	1500.06209	2052.44646	8368.22741	2434.98214	23.21637	1341.96887	4879.43655	11374.44583
18	746.47974	2939.60508	3224.15338	1255.21711	3112.84918	3601.12260	6659.65029	4010.79319	1742.24195	12053.30838	3448.32380	977.66272
19	4019.16272	938.92012	675.54466	4956.90995	815.69786	271.09981	10477.85364	147.22432	2293.10095	15752.81061	7182.04233	13630.76724
20	4054.92799	976.49731	710.14952	4993.52595	848.31893	310.79787	10518.18335	105.43719	2327.54033	15786.84801	7217.84940	13668.95889
21	3198.95045	133.15405	255.56799	4128.99140	277.35537	567.47511	9647.13895	798.04601	1494.83678	14944.46081	6359.73231	12794.32507
22	2363.35397	907.66717	1079.49294	3319.33668	931.73279	1502.86726	9031.40587	1851.40109	652.03451	14037.10965	5506.08745	12022.59049
23	3810.53653	558.27950	268.13376	4553.07080	396.71904	216.57793	10119.42666	538.99351	1878.78475	13336.31895	6772.82327	13238.13981
24	4132.67141	1059.04627	786.74907	5073.01636	920.97611	398.01384	10605.74019	16.15540	2402.67102	15860.50081	7295.42884	13751.81165
25	2980.94431	6055.18662	6304.46921	2119.48414	6168.34848	6720.20676	4697.63462	7103.62182	4687.14049	8773.49885	509.41731	6914.89342
26	4317.57073	1232.48083	979.97143	5251.93460	1221.89616	562.70393	10740.61688	224.86210	2597.57076	16055.55362	7480.14051	13916.75943
27	1542.01135	4069.98391	4381.40916	1121.20516	4259.39914	4720.10816	5498.41132	5133.74610	2920.81393	11182.54846	2709.32396	8713.57074
28	3160.46721	114.04824	297.66707	4089.03546	306.74582	610.30401	9601.85966	1021.83218	1461.21456	14907.15888	6320.78872	12752.01047
29	3359.14219	301.08305	278.78533	4284.39442	376.25922	435.95069	9773.92659	849.04820	1663.16662	15106.83938	6518.14007	12940.53550
30	3134.63283	99.96499	311.56701	4063.51043	307.76127	634.89290	9579.38490	1045.89722	1435.06864	14880.96428	6394.98880	1277.11684
31	1490.81085	1598.51337	1855.18517	2435.74321	1725.74940	2265.77823	8112.01375	2656.90553	269.94444	13230.29293	4653.77202	11132.93757
32	3322.63841	6405.06674	6858.96967	2427.49501	6524.40274	7072.06717	4335.62683	7459.54865	5042.22857	8431.34574	328.18745	6516.15792

Gambar 4.43 Jarak antar Data

Dari hasil tersebut dapat dilihat hasil perhitungan jarak antara data ke-1 dengan data ke-2 yaitu sebesar 3085.9, jarak antara data ke-1 dengan data ke-3 yaitu sebesar 3345.94, dimana jarak antara suatu data ke-i dengan data itu sendiri yaitu 0. Didapatkan plot dari jarak antar data yaitu:



Gambar 4.44 Hasil Plot

Dapat dilihat pada plot di atas pada plot adalah jarak paling rendah berwarna hijau dan jarak paling tinggi berwarna merah.

4. Membuat klasterisasi dari data tersebut menjadi 4 kelompok berdasarkan jarak antar data dengan syntax.

```
klaster <- kmeans(data_mina,centers=3, nstart = 25)  
klaster
```

Gambar 4.45 Membuat Klasterisasi 4 Kelompok

didapatkan hasil klaster yaitu:

```
> klaster  
K-means clustering with 3 clusters of sizes 3, 20, 10  
  
Cluster means:  
Tahanan.Dewasa Tahanan.Anak Napi.Dewasa Napi.Anak  
1 7217.667 96.33333 14707.0 186.33333  
2 659.950 9.50000 1565.0 29.10000  
3 3043.200 43.10000 6132.4 117.0000  
  
Clustering vector:  
[1] 3 2 2 3 2 2 3 2 2 1 3 1 2 3 2 3 2 2 2 2 2 3 2 3 2 2 2 2 3 1  
  
within cluster sum of squares by cluster:  
[1] 16430566 16098622 42172055  
(between_SS / total_SS =  89.5 %)  
  
Available components:  
[1] "cluster"      "centers"       "totss"        "withinss"      "tot.withinss" "betweenss"     "size"  
[8] "iter"          "ifault"
```

Gambar 4.46 Cluster

Didapatkan bahwa jika dilakukan klaster non-hierarki pada data tersebut tanpa scalling dengan 3 kelompok, maka didapatkan:

- Ukuran kelompok klaster yang didapatkan yaitu 3, 20 dan 10. Dimana untuk nilai rata-rata variabel tahanan dewasa pada kelompok 1 sebesar 7217.667,

untuk kelompok 2 nilai rata-ratanya sebesar 659.950, sedangkan pada kelompok 3 nilai rata-ratanya sebesar 3043.200. Untuk nilai rata-rata variabel tahanan anak pada kelompok 1 sebesar 96.333333, pada kelompok 2 sebesar 9.50000 dan pada kelompok 3 sebesar 43.10000.

- Nilai rata-rata variabel Napi Dewasa pada kelompok 1 sebesar 14707, pada kelompok 2 sebesar 1565 dan pada kelompok 3 sebesar 6132.4.
- Nilai rata-rata variabel Napi Anak pada kelompok 1 sebesar 186.3333, pada kelompok 2 sebesar 29.1000, dan pada kelompok 3 nilai rata-ratanya sebesar 117.
- Didapatkan bahwa anggota pada kelompok 1 yaitu data ke-10,12, dan data ke-33. Pada kelompok 3 anggotanya adalah data ke-1, 4, 7, 11, 14, 16, 18, 25, 27, dan data ke-32. Dan data sisanya termasuk ke kelompok 2.
- Dimana tahanan dewasa, tahanan anak, napi dewasa dan napi anak lebih banyak pada kelompok 1.
- Dimana nilai within cluster atau jarak antar objek didalam klaster yaitu untuk kelompok 1 yaitu sebesar 16430566, untuk kelompok 2 yaitu 16098622, dan untuk kelompok 3 yaitu 42172055
- Dimana hasil hitung between SS dibagi total SS yaitu 89.5

5. Membuat data frame hasil klaster.

```
k=data.frame(klaster$cluster)
view(k)
k
fviz_cluster(klaster, data= data_nina)
```

Gambar 4.47 Membuat Data Frame Hasil Klaster

6. Membandingkan hasil klaster jika dibuat jumlah kelompok pada klaster sebesar 2, 3, 4 dan 5 kelompok.

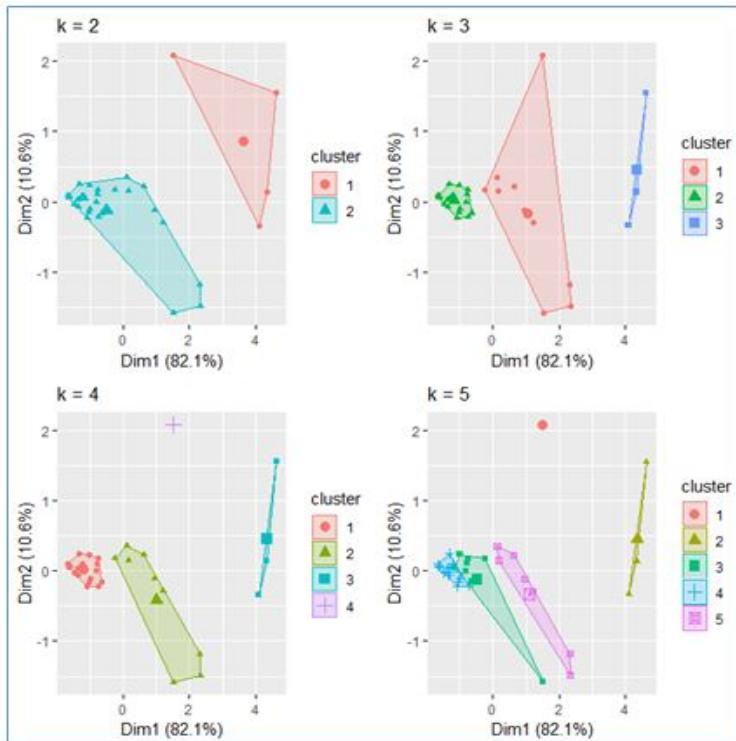
```
##comparing the result
k2 <- kmeans(data_nina, centers=2, nstart = 25)
k3 <- kmeans(data_nina, centers=3, nstart = 25)
k4 <- kmeans(data_nina, centers=4, nstart = 25)
k5 <- kmeans(data_nina, centers=5, nstart = 1)

p1 <- fviz_cluster(k2, geom="point", data= data_nina) + ggtitle("k = 2")
p2 <- fviz_cluster(k3, geom="point", data= data_nina) + ggtitle("k = 3")
p3 <- fviz_cluster(k4, geom="point", data= data_nina) + ggtitle("k = 4")
p4 <- fviz_cluster(k5, geom="point", data= data_nina) + ggtitle("k = 5")

library(gridExtra)
grid.arrange(p1,p2,p3,p4,nrow=2)
```

Gambar 4.48 Membandingkan Hasil Klaster

didapatkan hasil:



Gambar 4.49 Hasil Klaster

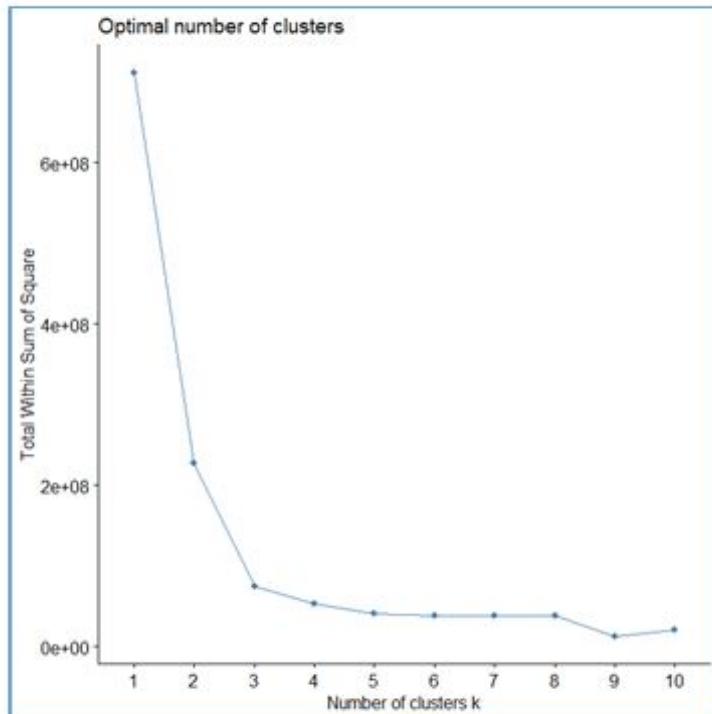
Pada hasil di atas dapat dilihat bahwa klaster yang memiliki nilai *within* antar data dalam kelompok paling kecil adalah dengan jumlah kelompok sebanyak 2, karena jarak pada satu titik dengan titik lainnya pada kelompok lebih dekat dibandingkan klaster dengan jumlah kelompok lainnya.

7. Menentukan jumlah kelompok klaster yang optimal dengan syntax:

```
fviz_nbclust(data_nina,kmeans,method="wss")  
fviz_nbclust(data_nina, kmeans, method="silhouette")
```

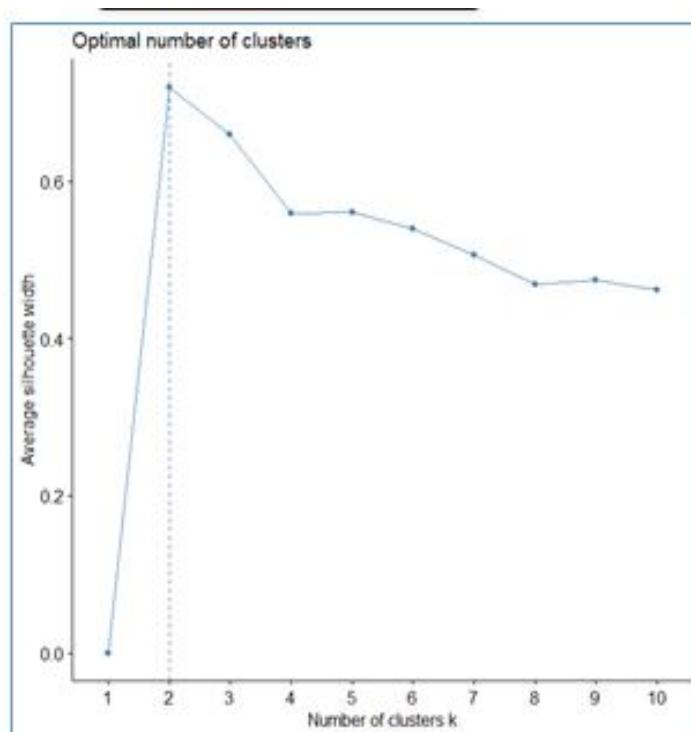
Gambar 4.50 Menentukan Jumlah Kelompok Klaster yang Optimal

didapatkan hasil



Gambar 4.51 Plot Optimal Number of Cluster

Pada plot tersebut dapat dilihat bahwa garis sudah mulai lamda pada jumlah kelompok sebanyak 2. Kemudian didapatkan juga output:



Gambar 4.52 Plot Optimal Number of Cluster

Dari plot di atas dapat dilihat bahwa garis paling tinggi adalah pada jumlah kelompok klaster sebanyak 2, sehingga jumlah kelompok sebanyak 2 merupakan jumlah klaster paling optimal.

4.2.9.2 Kmeans Clustering pada Python Data yang digunakan adalah data sekunder yaitu data Air Quality yang diambil dari <http://archive.ics.uci.edu/ml/datasets/Air+Quality>. Data ini berisi respons dari perangkat multisensor gas yang digunakan di lapangan di kota Italia. Rata-rata respons setiap jam dicatat bersama dengan referensi konsentrasi gas dari penganalisa bersertifikat.

1. Input Libraries

untuk menjalankan skrip berikut, kita memerlukan library pandas, numpy, matplotlib, dan scikit-learn.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
from sklearn.preprocessing import MinMaxScaler
```

Gambar 4.53 Input Libraries

2. Masukkan Data yang Akan di Kelompokkan

Setelah mengimpor semua dependensi yang diperlukan dalam tutorialini, selanjutnya kita akan memuat dataset.

```
air = pd.read_csv("AirQuality.csv")
air.head()
```

Gambar 4.54 Import Depedensi

	Date	Time	CO	PT08.S1	NMHC	C6H6	PT08.S2	Nox	PT08.S3(NOX)	NO2(GT)	PT08.S4(NO2)	PT08.S5(O3)	T	RH	AH	
0	10-03-04	18:00:00	2.6	1360	150	11.9	1046	166		1056	113	1692	1268	13.6	48.9	0.7578
1	10-03-04	19:00:00	2.0	1292	112	9.4	955	103		1174	92	1559	972	13.3	47.7	0.7255
2	10-03-04	20:00:00	2.2	1402	88	9.0	939	131		1140	114	1555	1074	11.9	54.0	0.7502
3	10-03-04	21:00:00	2.2	1376	80	9.2	948	172		1092	122	1584	1203	11.0	60.0	0.7867
4	10-03-04	22:00:00	1.6	1272	51	6.5	836	131		1205	116	1490	1110	11.2	59.6	0.7888

Gambar 4.55 Import Depedensi

3. Hilangkan Variabel yang Tidak Perlu

Kali ini gunakan variabel NMHC dan RH sehingga hilangkan variabel yang tidak dibutuhkan.

```
air =
air.drop(["Date","Time","CO","PT08.S1","C6H6","PT08.S2","AH","T","PT08.S5(O3)","PT08.S4(NO2)","NO2(GT)","PT08.S3(NOX)","Nox"], axis = 1)
air
```

Gambar 4.56 Variabel NMHC dan RH

	NMHC	RH
0	150	48.9
1	112	47.7
2	88	54.0
3	80	60.0
4	51	59.6
5	38	59.2
6	31	56.8
7	31	60.0
8	24	59.7
9	19	60.2
10	14	60.5

Gambar 4.57 Variabel NMHC dan RH

4. Preprocessing Data

Kita dapat melihat bahwa tidak semua nilai fitur bertipe sama. Beberapa dari mereka adalah angka dan beberapa dari mereka tidak. Untuk mempermudah perhitungan, kita akan mengumpulkan semua data numerik ke model. Mari kita lihat tipe data dari berbagai fitur yang dimiliki:

```
air.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9357 entries, 0 to 9356
Data columns (total 15 columns):
Date          9357 non-null object
Time          9357 non-null object
CO            9357 non-null float64
PT08.S1       9357 non-null int64
NMHC          9357 non-null int64
C6H6          9357 non-null float64
PT08.S2       9357 non-null int64
Nox           9357 non-null int64
PT08.S3(Nox)  9357 non-null int64
NO2(GT)       9357 non-null int64
PT08.S4(NO2)  9357 non-null int64
PT08.S5(O3)   9357 non-null int64
T              9357 non-null float64
RH             9357 non-null float64
AH             9357 non-null float64
dtypes: float64(5), int64(8), object(2)
memory usage: 1.1+ MB
```

Gambar 4.58 NMHC dan RH yang sudah bersifat numerik

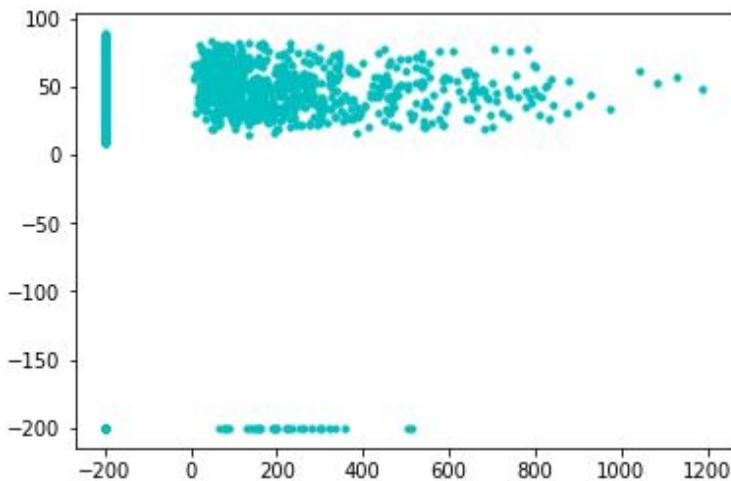
5. Visualisasi Data

Visualisasikan data yang ada dengan plot poin-poin dan periksa apakah kita dapat melihat kelompok manapun dengan syntax seperti berikut.

```
plt.scatter(air.NMHC , air.RH, s =10, c = "c", marker = "o", alpha =
1)
plt.show()
```

Gambar 4.59 Visualisasi Data

Sehingga didapatkan hasil visualisasi data seperti gambar berikut.



Gambar 4.60 Visualisasi

6. Buat Clusters

Untuk membuat sebuah K-means cluster ikuti langkah berikut ini.

Menentukan variabel yang akan dikelompokkan. Disini variabel yang akan dikelompokkan adalah variabel NMHC dan variabel RH.

```
air_x = air.iloc[:, 0:2]
air_x.head()
```

Gambar 4.61 Visualisasi

	NMHC	RH
0	150	48.9
1	112	47.7
2	88	54.0
3	80	60.0
4	51	59.6

Gambar 4.62 Visualisasi

Mengubah variabel data frame menjadi array sebelum menentukan nilai K

```
x_array = np.array(air_x)
print(x_array)
```

```
[[ 150.    48.9]
 [ 112.    47.7]
 [  88.    54. ]
 ...
 [-200.   18.3]
 [-200.   13.5]
 [-200.   13.1]]
```

Gambar 4.63 Array

Menstandarkan kembali ukuran variabel array.

```
scaler = MinMaxScaler()  
x_scaled = scaler.fit_transform(x_array)  
x_scaled
```

```
array([[0.25197984, 0.86214063],  
       [0.22462203, 0.85798407],  
       [0.20734341, 0.87980603],  
       ...,  
       [0.          , 0.75614825],  
       [0.          , 0.739522  ],  
       [0.          , 0.73813647]])
```

Gambar 4.64 Array

Menentukan dan konfigurasi fungsi k-means. syntax berikut merupakan syntax untuk mengelompokkan data menjadi 5 cluster

```
kmeans = KMeans(n_clusters = 5, random_state=123)
```

Gambar 4.65 Konfigurasi Fungsi Kmeans Cluster

menentukan nilai K (cluster) dari data.

```
kmeans.fit(x_scaled)
```

```
KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,  
n_clusters=5, n_init=10, n_jobs=1, precompute_distances='auto',  
random_state=123, tol=0.0001, verbose=0)
```

Gambar 4.66 Penentuan Nilai K

Menampilkan pusat cluster. Untuk melihat pusat cluster, ikuti langkah berikut.

```
print(kmeans.cluster_centers_)
```

```
[[ -1.26634814e-15  9.12226444e-01]  
 [ 2.25995035e-02 -5.55111512e-16]  
 [ 2.38485225e-01  8.69034622e-01]  
 [-1.10328413e-15  8.11122325e-01]  
 [ 5.55427384e-01  8.50556570e-01]]
```

Gambar 4.67 Pusat Cluster

Menambahkan kolom cluster dalam data frame ritel.

```
air["kluster"] = kmeans.labels_
```

Gambar 4.68 Tambah Kolom

Memvisualisasikan hasil cluster

```
output = plt.scatter(x_scaled[:,0], x_scaled[:,1], s = 100, c =
air.kluster, marker = "o", alpha = 1, )

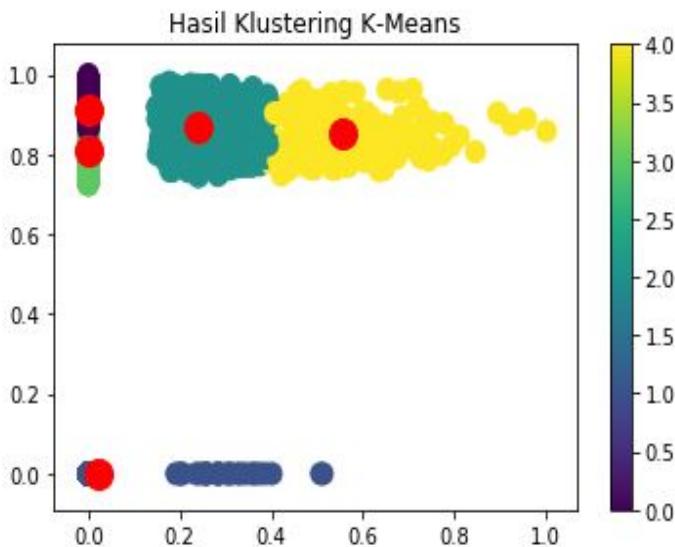
centers = kmeans.cluster_centers_
plt.scatter(centers[:,0], centers[:,1], c='red', s=200, alpha=1 ,
marker="o");

plt.title("Hasil Klustering K-Means")
plt.colorbar (output)

plt.show()
```

Gambar 4.69 Visualisasi Hasil Cluster

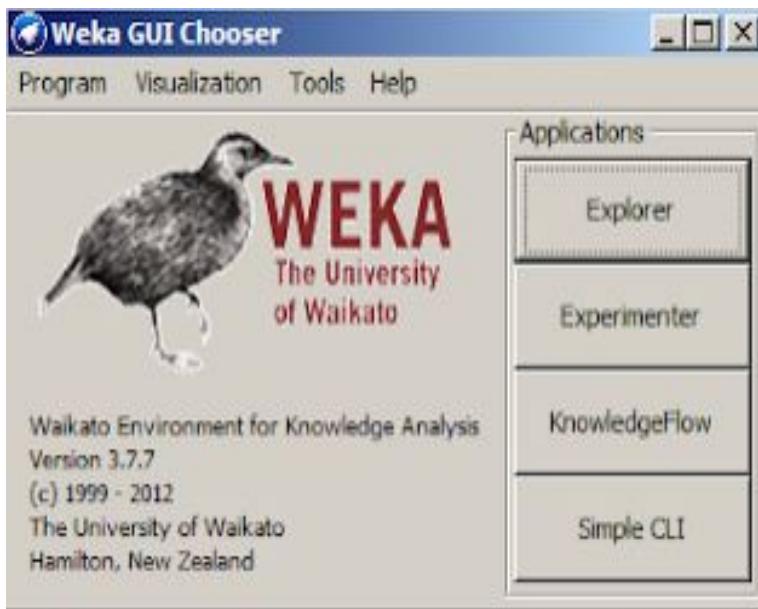
Sehingga didapatkan hasil output seperti pada gambar berikut.



Gambar 4.70 Visualisasi Hasil Cluster

Dalam kasus ini, respons NMHC dan respons RH dari perangkat multisensor gas yang digunakan di lapangan di kota Italia terdapat 5 cluster dan terdapat 5 titik (ditampilkan dengan warna merah) yang menjadi pusat data setiap cluster.

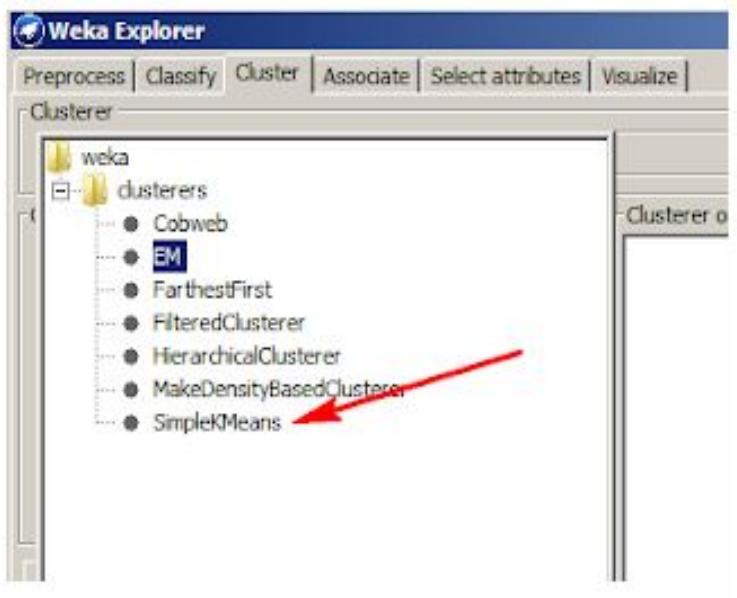
4.2.9.3 WEKA API kluster menggunakan KMeans Langkah untuk melakukan Kmeans pada weather.arff :



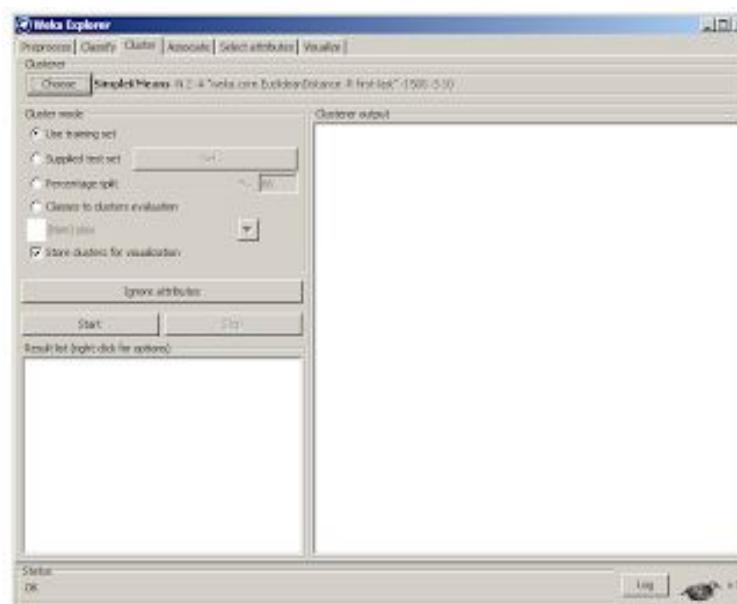
Gambar 4.71 GUI yang disediakan WEKA



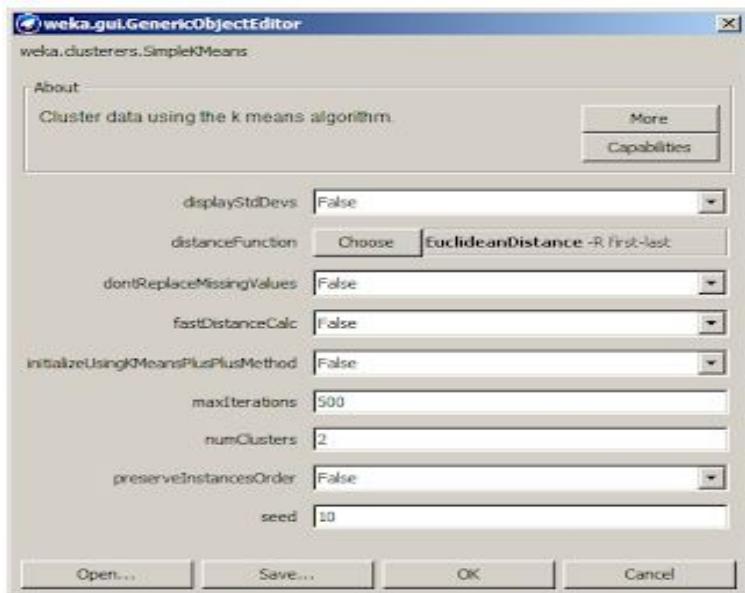
Gambar 4.72 Ketika weather.arff sudah dimasukkan



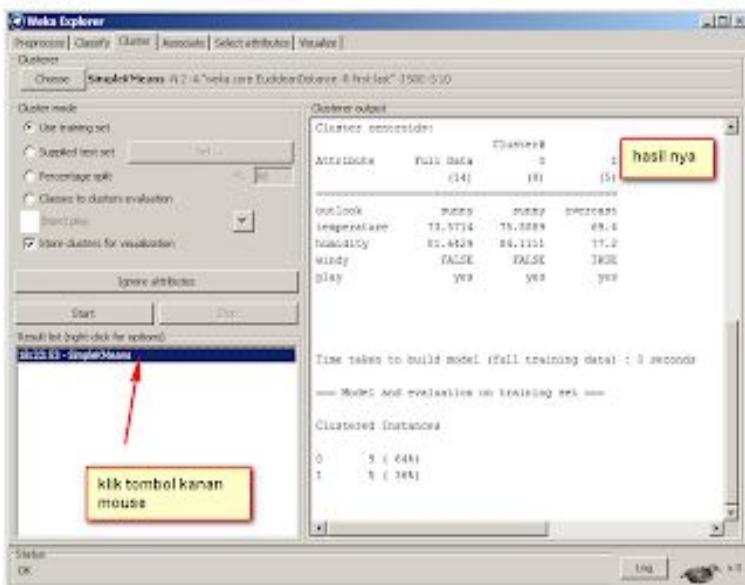
Gambar 4.73 Memilih SImpleKMeans



Gambar 4.74 Tampilan SimpleKMeans sudah siap untuk mulai



Gambar 4.75 Setingan tambahan jika diinginkan



Gambar 4.76 Hasil Clusternya

```

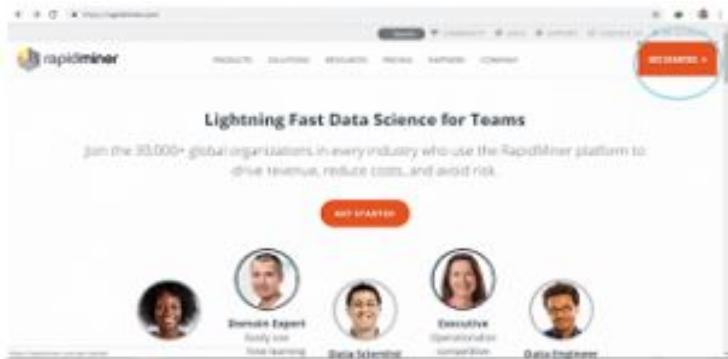
11 @data
12 0,sunny,85,85, FALSE,no,cluster0
13 1,sunny,80,90, TRUE,no,cluster0
14 2,overcast,83,86, FALSE, yes,cluster0
15 3,rainy,70,96, FALSE, yes,cluster0
16 4,rainy,68,80, FALSE, yes,cluster0
17 5,rainy,65,70, TRUE,no,cluster1
18 6,overcast,64,65, TRUE, yes,cluster1
19 7,sunny,72,95, FALSE,no,cluster0
20 8,sunny,69,70, FALSE, yes,cluster0
21 9,rainy,75,80, FALSE, yes,cluster0
22 10,sunny,75,70, TRUE, yes,cluster1
23 11,overcast,72,90, TRUE, yes,cluster1
24 12,overcast,81,75, FALSE, yes,cluster0
25 13,rainy,71,91, TRUE,no,cluster1

```

Gambar 4.77 Cluster yang terbentuk per instance nya

4.2.9.4 Clustering Dengan Algoritma K-Means Di RAPIDMINER Tutorial menggunakan rapid miner untuk menghitung K-Means pada TextProcessing untuk pemula dengan pemahaman yang mudah.

1. Download RapidMiner



Gambar 4.78 Tampilan Download Rapidminer

Pilih GetStarted yang terdapat pada layar kanan anda. Setelah itu anda diharuskan untuk mengisi email yang akan anda gunakan untuk registrasi. Usahakan email yang anda isikan masih aktif lalu klik download seperti gambar

dibawah ini.

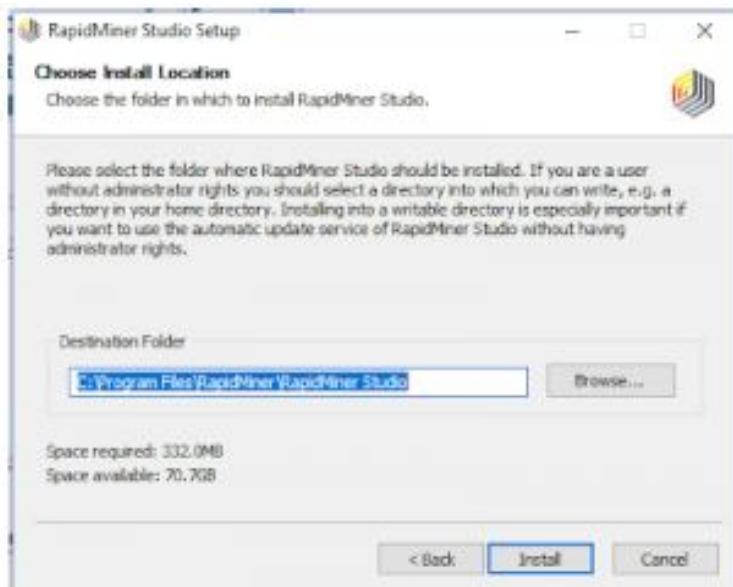
2. Instalasi RapidMiner

Setelah kita berhasil mendownload software yang diperlukan, maka step selanjutnya ada melakukan instalasi.



Gambar 4.79 Install Rapidminer

Buka file yang telah kita download tadi, lalu klik 2x pada file tersebut.

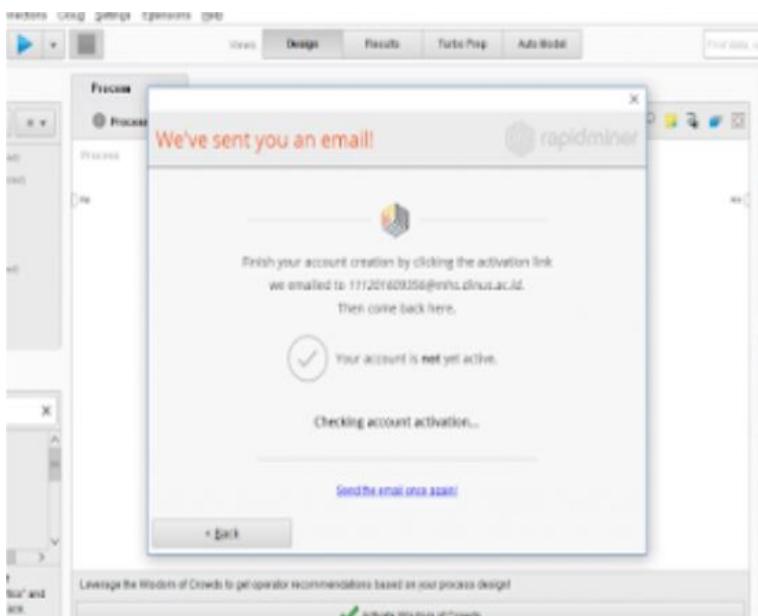


Gambar 4.80 Halaman Instalasi

Pilih next hingga muncul tampilan diatas. Anda dapat memilih dimana file installation RapidMiner akan ditempatkan. Setelah selesai, silahkan tekan tombol install. Tunggu instalasinya selesai, hingga muncul tulisan Finish. Setelah instalasi selesai, maka RapidMiner siap untuk digunakan.

3. Membuka dan Menjalankan RapidMiner

Setelah proses instalasi berjalan dengan lancar, kita langsung saja membuka RapidMiner. Untuk tampilan awal/pertama kali kita menjalankan software tersebut kita akan diminta untuk membuat sebuah akun agar dapat menggunakannya. Pilih saja dibagian Educational.



Gambar 4.81 Proses menjalankan RapidMiner

isi data diri anda dengan selengkap-lengkapnya lalu click create my account.
Maka akan muncul tampilan seperti dibawah ini

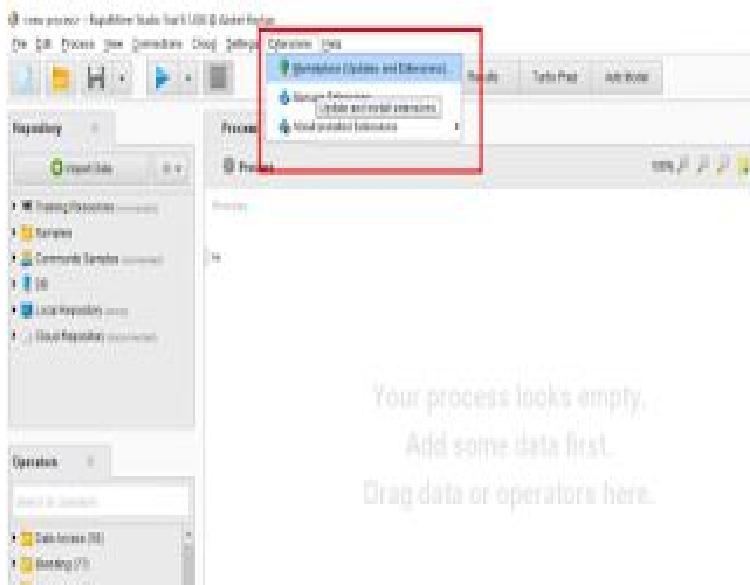


Gambar 4.82 Pengisian Data

Langkah selanjutnya yaitu buka email yang telah didaftarkan, maka anda akan memperoleh email rapidminer, klik dan didalam isi email itu klik Confirm your email address. Setelah proses aktivasi selesai, RapidMiner siap dijalankan.

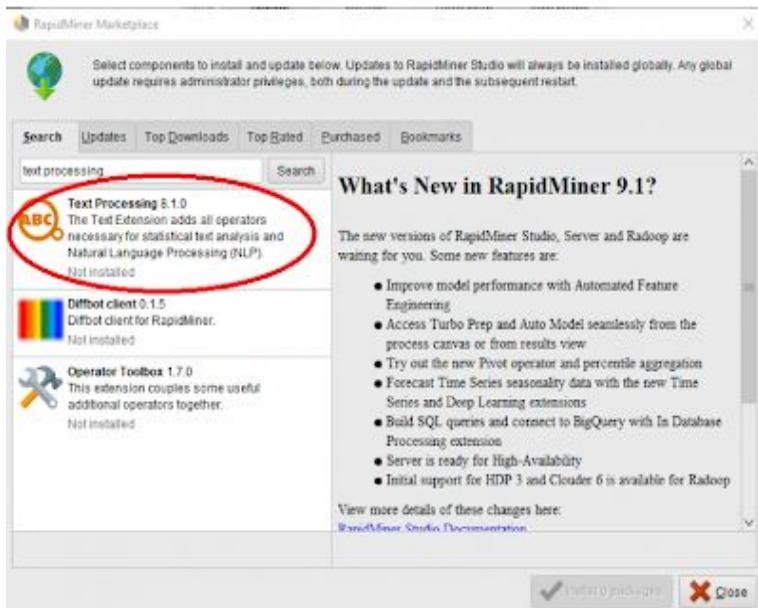
4. Menghitung K-Means

Langkah awal anda dapat masuk ke menu extensions lalu pilih marketplace seperti gambar dibawah ini.



Gambar 4.83 Proses *Kmeans*

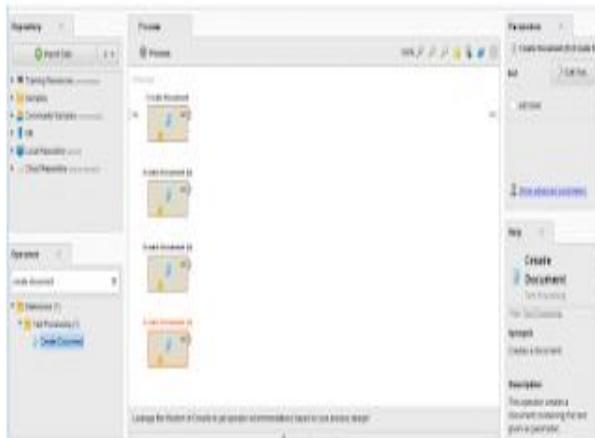
Search *Text Processing* dan pilih *extension* seperti lingkaran dibawah ini.



Gambar 4.84 *Text Processing*

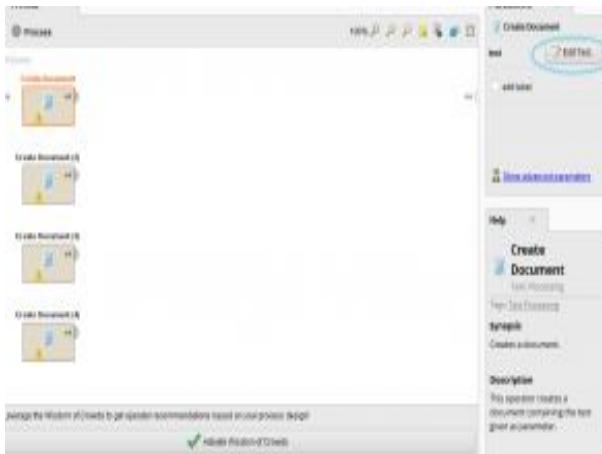
Install packages yang anda pilih dan tunggu hingga selesai lalu *restart Rapid-Miner* anda. Setelah proses *Extension Instalasi* berjalan dengan baik, langsung lanjut ke step berikutnya

- Search pada bagian operator di kanan bawah Rapid miner dengan create document. masukan operator ini sesuai dengan dokumen yang ingin anda proses. Pada kali ini, terdapat 4 Dokumen.



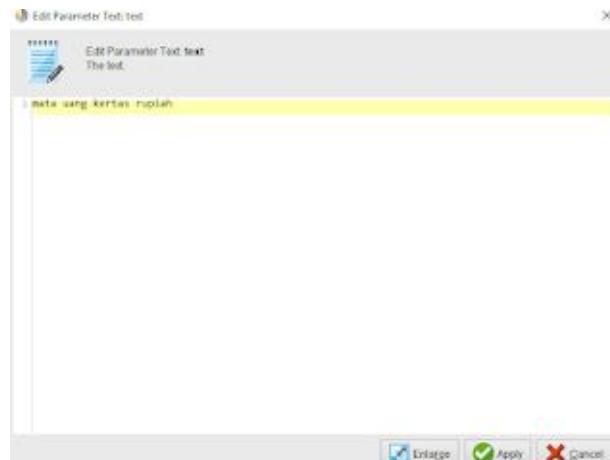
Gambar 4.85 Create Document

- Isikan tiap dokumen dengan klik create document pertama dan klik edit text di bagian Parameters sebelah kanan atas.



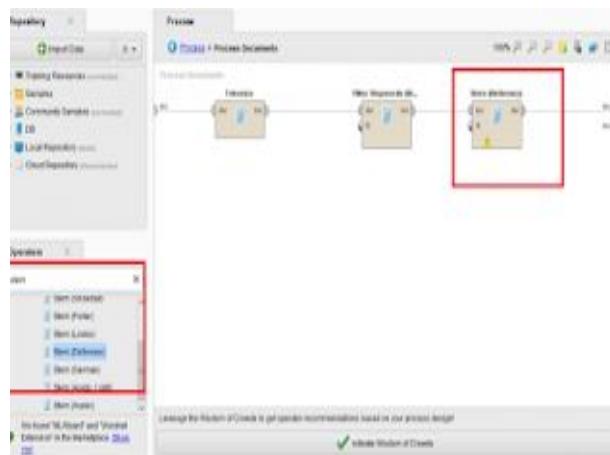
Gambar 4.86 Edit Text

- Isikan *document* sesuai yang anda ingin analisis. Lalu klik *apply*.

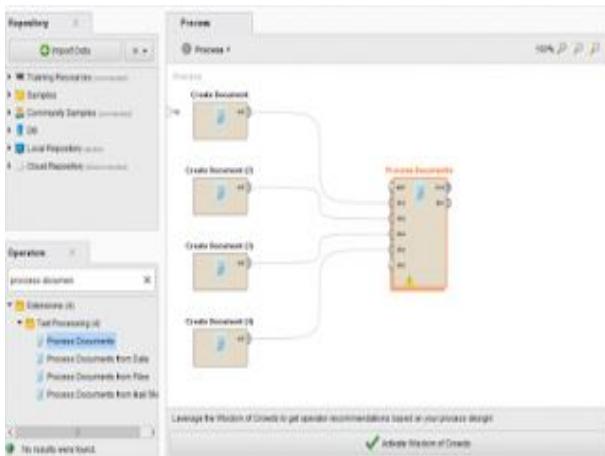


Gambar 4.87 Pengisian Dokumen

- lakukan hal yang sama dengan document yang lainnya.
- Pada operator cari proccess document lalu masukan dalam project. Lalu sambungkan tiap document tadi kedalam proccess document seperti digambar

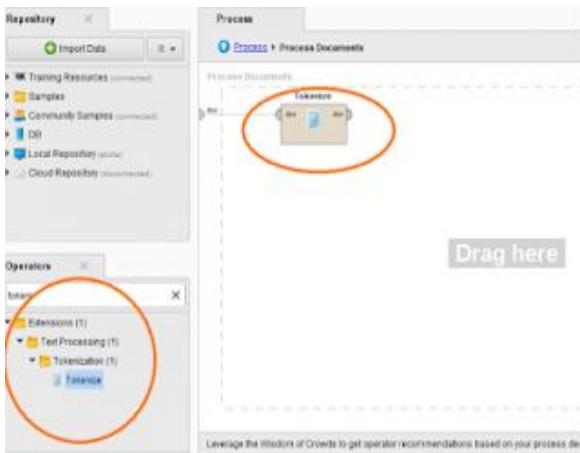


Gambar 4.88 Proccess Document



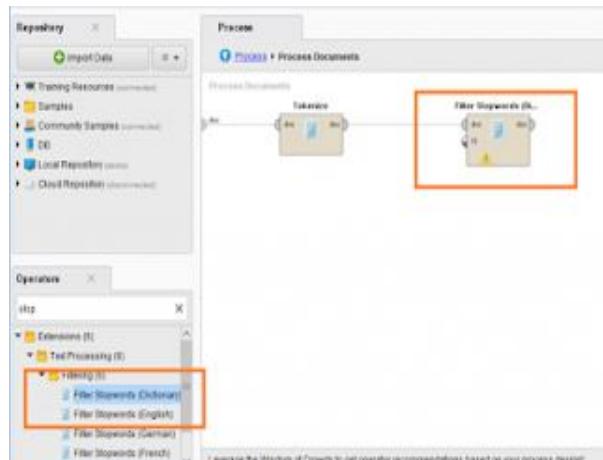
Gambar 4.89 *Proccess Document*

- Klik 2x pada proccess document, lalu isikan operator tokenize didalamnya.



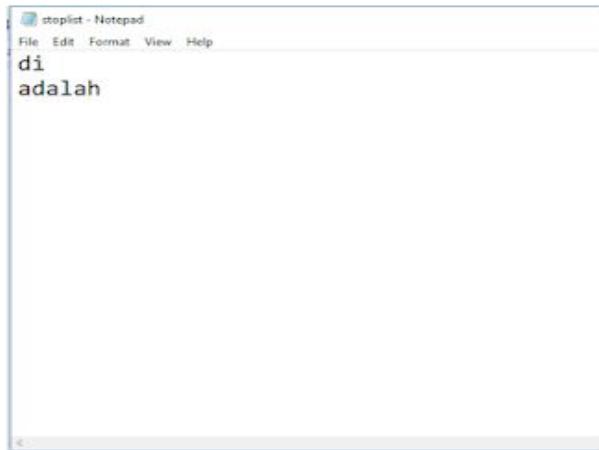
Gambar 4.90 *Isikan Operator Tokenize*

- Tambahkan operator stopword setelah tokenize. pilihlah *Stopwords(Dictionary)*.



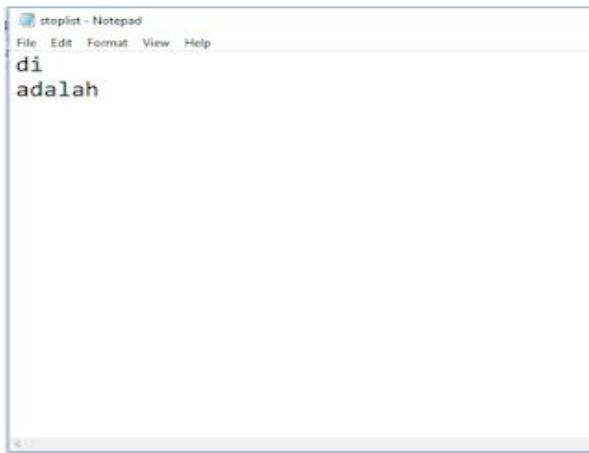
Gambar 4.91 Operator Stopword

- Stopword memerlukan file sebagai rujukan kata mana saja yang akan dihapuskan oleh proses. Dalam hal ini saya menggunakan *notepad* sebagai *database* dan mengisikan kata seperti dibawah ini.



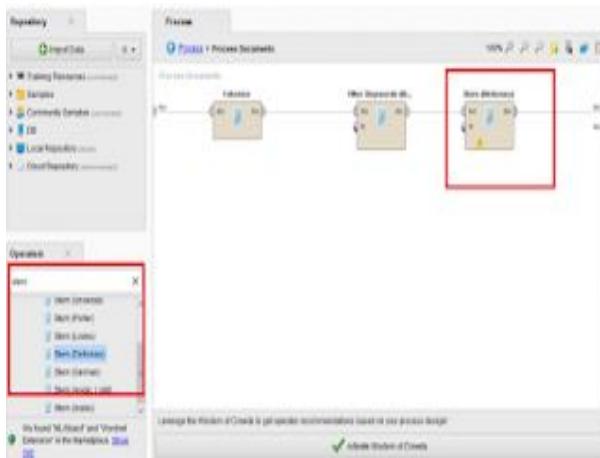
Gambar 4.92 Penggunaan Notepad sebagai Database

- Langkah selanjutnya pada parameters Stopword silahkan anda masukan file yang telah anda buat tadi. Lalu klik open.



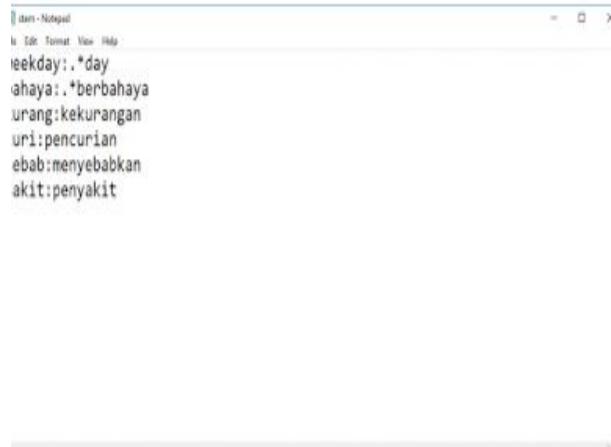
Gambar 4.93 masukan file pada parameter stopword

- Selanjutnya masukan operator Stem(Dictionary) kedalam process.



Gambar 4.94 masukan operator stem

- Sama halnya dengan Stopword, Stem juga perlu file. disini saya sudah menyimpan file bertipe txt seperti digambar.



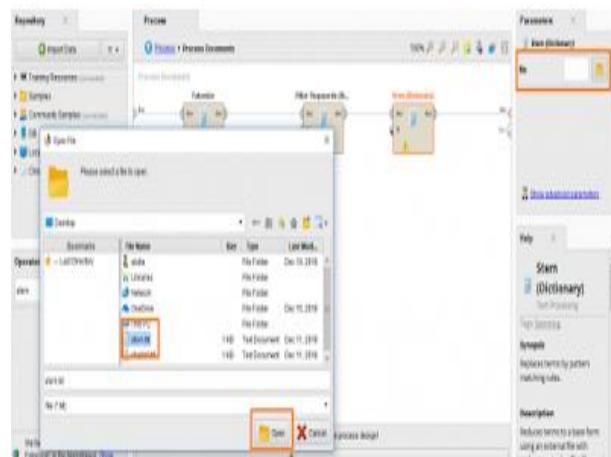
```

stem - Notepad
File Edit Format View Help
weekday:.*day
ahaya:.*berbahaya
urang:kekurangan
uri:pencurian
ebab:menyebabkan
akit:penyakit

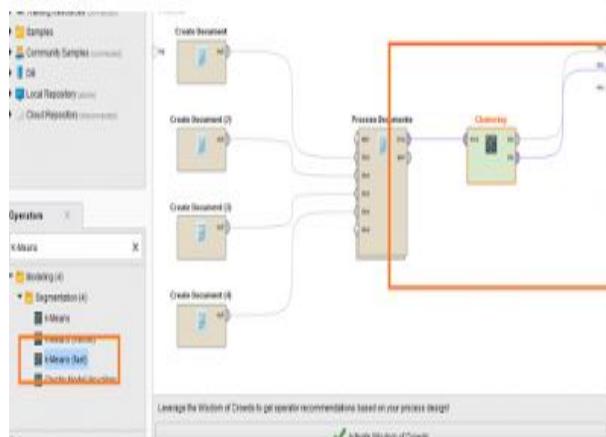
```

Gambar 4.95 File Stem

- Masukan file kedalam process stem sama seperti stopword.

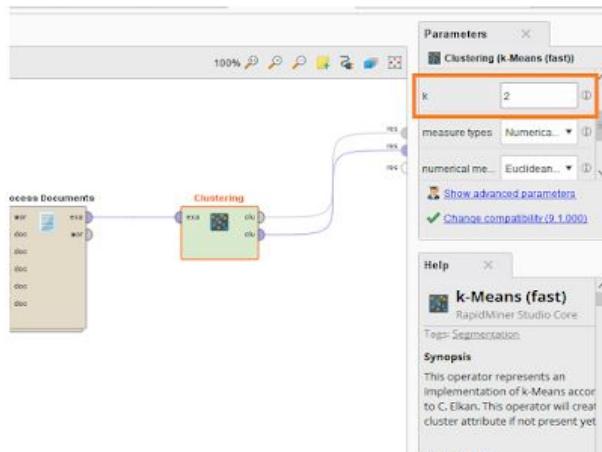
**Gambar 4.96 File Stem**

- Kembali ke halaman procces awal. Pada operator, search k-means lalu pilih *K-Means(Fast)*.



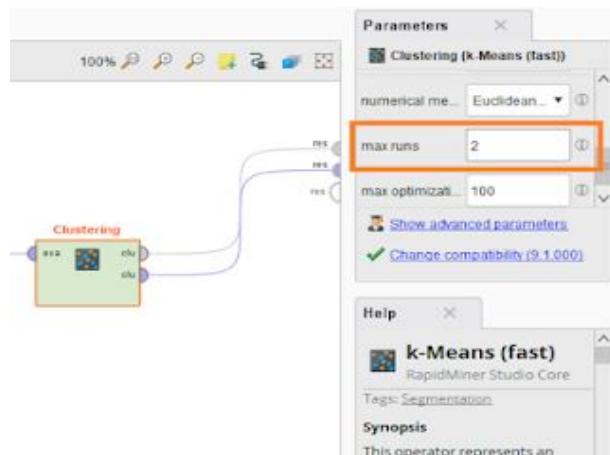
Gambar 4.97 File Stem

- Kita akan mengatur parameter *K-means* disebelah kanan atas layar anda. Atur K sebagai Centroid yang akan anda inginkan. dalam hal ini saya ingin ada 2 centroid maka saya isi 2.



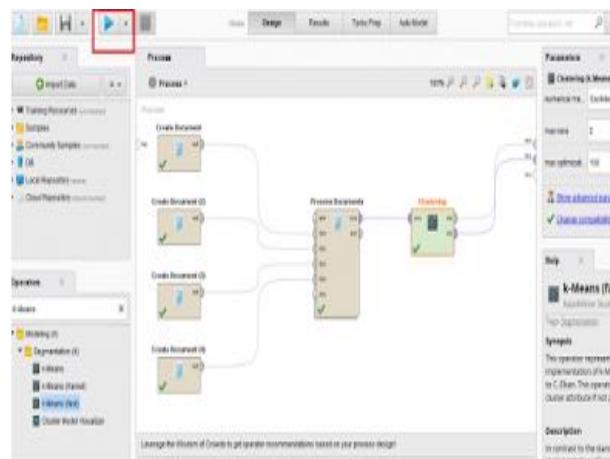
Gambar 4.98 Parameter *K-means Clustering*

- Scroll parameters hingga anda menemukan max runs. Max Runs adalah maksimal iterasi akan dilakukan. dalam hal ini saya mengisikan 2 iterasi.



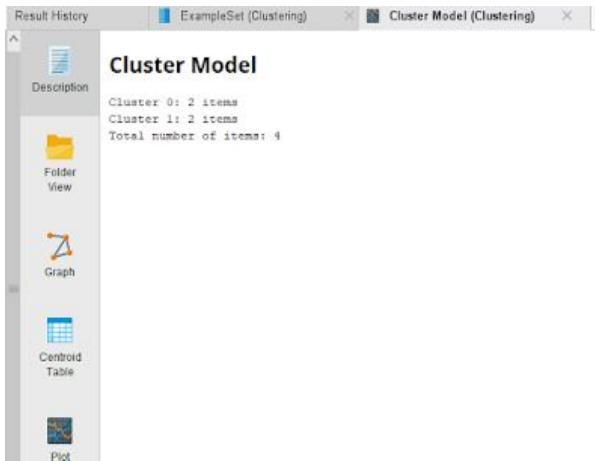
Gambar 4.99 Maxruns

- Silahkan run dengan klik gambar play warna biru dilayar anda seperti digambar.



Gambar 4.100 Proses Cluster

- Proses akan menghasilkan 2 hasil yaitu Cluster Model dan ExampleSet. Didalam cluster model akan ditunjukkan 2 cluster yaitu Cluster 0 dan Cluster 1, dan masing-masing cluster memiliki 2 items atau 2 dokumen.



Gambar 4.101 Cluster Model dan ExampleSet

- Sementara di ExampleSet anda akan ditunjukan Dokument dengan Id tertentu akan masuk kedalam cluster apa. Dan didalam sini juga kita dapat melihat TfIdf yang dihasilkan oleh term-term yang ada didalam dokumen.

Item No.	ID	cluster	salami	rind	ketchup	bacon	lettuce	meatball	ribs
1	1	cluster_1	0	0	0.001	0.001	0	0	0.001
2	2	cluster_1	0.001	0	0.001	0	0	0	0.001
3	3	cluster_2	0	0.001	0	0	0	0.001	0
4	4	cluster_2	0	0.01	0	0	0.001	0.001	0

Gambar 4.102 ExampleSet

- Dilihat dari gambar diatas, Dokumen 1 dan 2 menjadi 1 Cluster atau 1 kelompok. Dan dokumen 3 serta 4 menjadi 1 cluster.
- Dapat diartikan bahwa dokumen 1 memiliki kemiripan dengan dokumen 2 dan dokumen 3 mirip dengan dokumen 4.

4.2.10 Pengaduan

Perilaku pengaduan konsumen (*consumer complaint behavior*) terdiri dari semua tindakan konsumen yang dilakukan sebagai akibat ketidakpuasannya terhadap pembelian. Konsumen yang mengajukan keluhan karena ada perasaan kecewa atau tidak puas terhadap pelayanan yang diberikan oleh produsen. ada beberapa jenis perilaku komplain yaitu niat untuk complain, *Word of mouth negatif*, dan niat untuk beralih

Sikap terhadap mengeluh didefinisikan sebagai kecenderungan ketidakpuasan konsumen secara pribadi untuk mencari kompensasi dari perusahaan. Sikap ini dikonseptualisasikan sebagai pengaruh "kebaikan" atau "keburukan" secara keseluruhan dari mengeluh kepada penjual dan tidak spesifik untuk bagian tertentu dari ketidakpuasan.

Persepsi peluang suksesnya keluhan didefinisikan sebagai persepsi peluang mendapatkan hadiah seperti pengembalian dana, pertukaran, atau permintaan maaf melalui mengeluh kepada perusahaan. Jika konsumen percaya bahwa mengeluh kepada perusahaan sangat berperan dalam mencapai beberapa konsekuensi yang diinginkan dan konsekuensi ini dianggap untuk memberikan nilai yang diinginkan, konsistensi kognitif akan memotivasi konsumen untuk terlibat dalam niat keluhan yang lebih tinggi dan lebih lanjut.

4.2.11 Telkomsel

Telkomsel Sejak berdiri pada tanggal 26 Mei 1995, Telkomsel secara konsisten melayani negeri, menghadirkan akses telekomunikasi kepada masyarakat Indonesia yang tersebar dari Sabang sampai Merauke.

Saat ini Telkomsel adalah operator selular terbesar di Indonesia dengan 168 juta pelanggan dan untuk melayani pelanggannya yang tersebar di seluruh Indonesia, termasuk juga di daerah terpencil dan pulau terluar serta daerah perbatasan negara, Telkomsel menggelar lebih dari 204 ribu BTS.

Telkomsel secara konsisten mengimplementasikan teknologi seluler terkini dan menjadi yang pertama meluncurkan secara komersial layanan mobile 4G LTE di Indonesia. Memasuki era *digital*, Telkomsel terus mengembangkan bisnis *digital*, diantaranya *Digital Advertising*, *Digital Lifestyle*, *Mobile Financial Services*, dan *Internet of Things*. Untuk melayani kebutuhan pelanggan, Telkomsel menggelar *call center* 24 jam dan layanan GraPARI yang tersebar di seluruh Indonesia.

Untuk memberikan layanan yang prima kepada masyarakat di dalam menikmati gaya hidup *digital* (*digital lifestyle*), Telkomsel turut membangun ekosistem digital di tanah air melalui berbagai upaya pengembangan DNA (*Device, Network dan Applications*), yang diharapkan akan mempercepat terbentuknya masyarakat digital

Indonesia. Selain itu Telkomsel juga aktif mendorong generasi muda untuk secara positif menggunakan teknologi.

Telkomsel akan selalu hadir untuk menginspirasi masyarakat dengan memanfaatkan teknologi terdepan, produk dan layanan yang kompetitif, serta solusi inovatif. Hal ini akan mengantarkan Indonesia menuju perekonomian masyarakat berbasis *broadband* sesuai *roadmap* teknologi selular. Kecintaan pada negeri mendorong Telkomsel untuk terus berkreasi menghadirkan layanan telekomunikasi terbaik bagi masyarakat Indonesia.

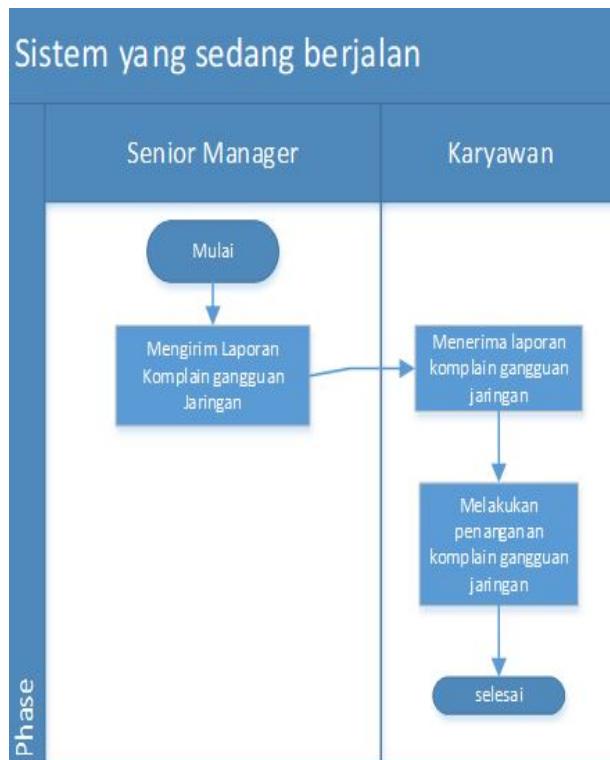
4.2.12 Jenis-jenis gangguan pada Jaringan Telkomsel

1. FO (*Fiber Optic*) kabel yang berfungsi untuk menyambungkan jaringan
2. Assesoris adalah jaringan yang digunakan pada saat pemasangan pada *fiber optic*
3. Cada adalah *power* yang menghasilkan energi listrik yang berguna sebagai sumber listrik
4. Modul adalah penamaan khusus untuk *radio ip*
5. Metro adalah perangkat *layer 2/3* yang berfungsi sebagai *router*
6. Logic adalah konfigurasi atau *setting logic* di perangkat metro
7. ONT adalah perangkat atau terminal yang ada di lapangan

4.3 Analisis

Analisis merupakan kegiatan penguraian suatu sistem yang utuh dan nyata kedalam bagian atau komponen yang bertujuan untuk mengidentifikasi serta mengevaluasi masalah-masalah yang ada, yang bertujuan untuk mempelajari aktivitas system untuk mendapat gambaran yang menyeluruh dan permasalahan yang terjadi serta kebutuhan-kebutuhannya.

4.3.1 Analisis Sistem Yang Sedang Berjalan

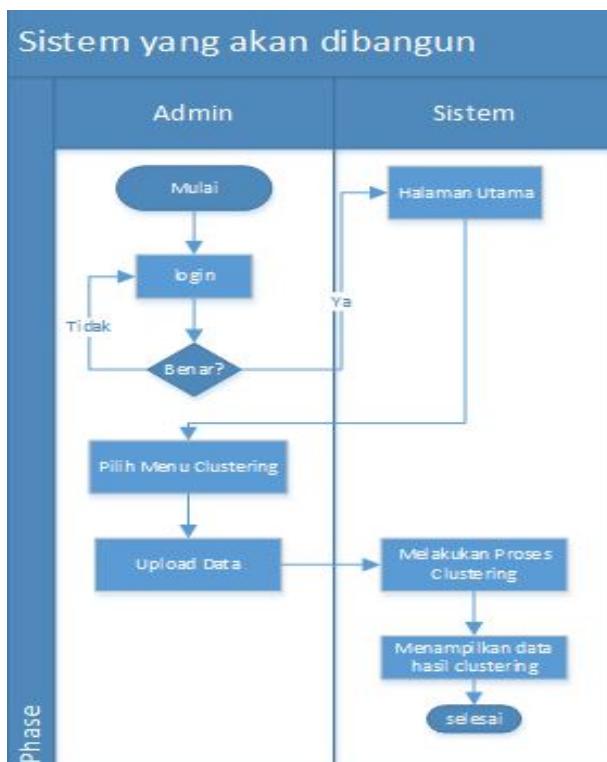


Gambar 4.103
Flowmap yang sedang berjalan

Keterangan :

1. Senior Manager mengirimkan laporan komplain gangguan jaringan kepada karyawan atau staff
2. Kemudian setelah karyawan menerima laporan komplain, selanjutnya melakukan penanganan komplain gangguan jaringan.

4.3.2 Analisis Sistem Yang Akan Dibangun



Gambar 4.104
Flowmap yang akan dibangun

Keterangan :

1. Admin login dengan memasukkan Username dan password.
2. Apabila Username dan password benar, maka admin akan masuk ke halaman utama, jika admin salah memasukkan Username dan password, admin akan kembali lagi ke menu login.
3. Setelah dari halaman utama, admin akan upload data yang akan diolah oleh sistem dengan menggunakan metode K-Means Clustering.
4. Kemudian data yang telah di upload akan diproses dalam sistem.
5. Setelah dilakukan pemrosesan data, maka sistem akan mengeluarkan hasil dari pengklasteran.

4.3.3 Analisis Perangkat yang Digunakan

Perangkat yang digunakan merupakan suatu cara agar dapat menghasilkan aplikasi yang sesuai dengan perangkat yang akan digunakan. Perangkat yang digunakan terdiri dari:

1. Perangkat Lunak (*Software*)

Adapun spesifikasi perangkat keras (Hardware) yang digunakan sebagai berikut:

No	Nama Perangkat	Spesifikasi	Keterangan
1	Memory	4 GB RAM	<i>Memory System</i> yang digunakan
2	Processor	<i>Intel Core™ i3</i>	Untuk kecepatan transfer data dari sistem yang sangat bergantung pada kecepatan prosesor komputer
3	System Type	64-bit <i>Operating System</i> , <i>x64-Based Processor</i>	

Gambar 4.105 Spesifikasi Perangkat keras

2. Perangkat Lunak (*Software*)

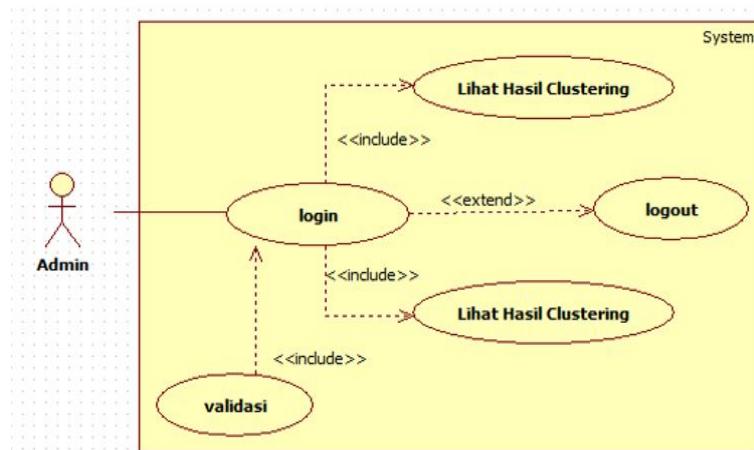
No	Tools / Software	Fungsi
1.	Windows 10	Sistem Operasi
2.	MySQL	Server Basis Data
3.	Google Chrome	Browser

Gambar 4.106 Spesifikasi Perangkat Lunak

4.4 Perancangan

4.4.1 Use Case Diagram

Use case diagram menggambarkan alur sistem secara ringkas dan menggambarkan kebutuhan fungsionalitas yang diharapkan oleh sebuah sistem. Berikut use case diagram yang digunakan:



Gambar 4.107 Usecase Diagram

4.4.2 Definisi Use Case

No	Use case	Deskripsi
1	Upload File	Merupakan langkah untuk mengolah data gangguan jaringan.
2	Lihat hasil clustering	User akan melihat hasil klasterisasi durasi penanganan gangguan jaringan baik itu cepat, sedang, ataupun lambat.
3	Login	Merupakan proses untuk melakukan identifikasi pengguna system dengan memasukkan Username dan password
4	Validasi	Validasi merupakan proses pengecekan hak akses kepada siapa saja yang berhak mengakses sistem yang akan dibangun.
5	Logout	Merupakan proses untuk keluar dari sistem sebagai pengguna sistem.

Gambar 4.108 Definisi use case

4.4.3 Skenario use case

Identifikasi	
Nomor	01
Nama	Kelola data complain gangguan jaringan
Tujuan	Merupakan langkah untuk mengolah data yaitu dengan cara upload data
Deskripsi	
Aktor	Admin
Scenario utama	
Kondisi awal	Masuk form login
Aksi aktor	Reaksi system
1. Admin memilih menu data	2. Sistem akan menampilkan tampilan untuk memasukkan data yang nantinya akan diupload
3. Admin memasukkan data complain gangguan	4. Sistem akan melakukan verifikasi data dan menyimpan data gangguan di dalam database
kondisi akhir	Kelola data complain gangguan berhasil

Gambar 4.109 Skenario use case

4.4.4 Skenario Login

<u>Identifikasi</u>	
<u>Nomor</u>	03
<u>Nama</u>	<i>Login</i>
<u>Tujuan</u>	Untuk Masuk kedalam SIstem
<u>Deskripsi</u>	
<u>Aktor</u>	<i>User</i>
<u>Scenario utama</u>	
<u>Kondisi awal</u>	<i>Masuk form login</i>
<u>Aksi actor</u>	<i>Reaksi system</i>
1. User memasukkan Username dan password	2. system akan menerima masukan Username dan password
3. User harus menekan tombol login	4. sistem akan melakukan validasi Username dan password. Apabila Username dan password benar maka akan muncul halaman utama. Jika salah, maka system akan memunculkan kotak pesan
<u>kondisi akhri</u>	<i>Jika Username dan password yang dimasukkan oleh User tidak sesuai maka akan muncul kotak pesan.</i>

Gambar 4.110 Skenario Login

4.4.5 Skenario *Validasi*

<u>Identifikasi</u>	
<u>Nomor</u>	04
<u>Nama</u>	<i>Validasi</i>
<u>Tujuan</u>	Untuk memastikan data yang dimasukkan telah benar, agar data yang kurang atau salah dapat diperbaiki sebelum disimpan.
<u>Deskripsi</u>	
<u>Aktor</u>	Admin dan <i>User</i>
<u>Scenario utama</u>	
<u>Kondisi awal</u>	Masuk <i>form login</i>
<u>Aksi actor</u>	Reaksi <i>system</i>
1. <i>User</i> memasukkan data	2. <i>system</i> akan menerima masukan data
3. <i>User</i> harus menekan tombol <i>login</i> atau tombol simpan	4. <i>sistem</i> akan mem <i>validasi</i> data yang baru dimasukkan . Apabila data benar maka akan muncul halaman selanjutnya. Jika salah, maka <i>system</i> akan memunculkan kotak pesan
<u>kondisi akhri</u>	jika data yang dimasukkan oleh <i>User</i> tidak sesuai maka akan muncul kotak pesan.

Gambar 4.111 Skenario *Validasi*

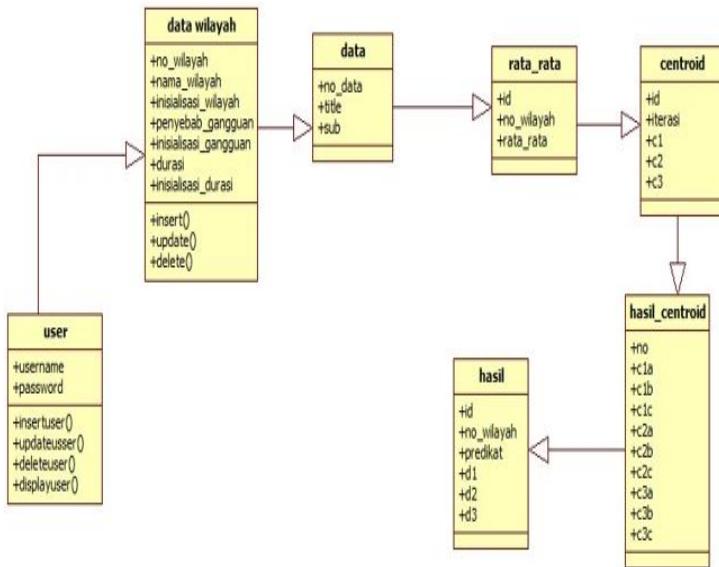
4.4.6 Skenario Logout

Identifikasi	
Nomor	05
Nama	<i>Logout</i>
Tujuan	Untuk mengeluarkan akun pengguna ketika pengguna telah selesai menggunakan sistem
Deskripsi	
Aktor	Admin dan User
Scenario utama	
Kondisi awal	Menekan tombol <i>logout</i>
Aksi actor	Reaksi <i>system</i>
1. <i>User</i> menekan tombol <i>logout</i>	2. <i>system</i> akan mengeluarkan akun pengguna
kondisi akhri	<i>User</i> akan kembali ke halaman login

Gambar 4.112 Skenario Log out

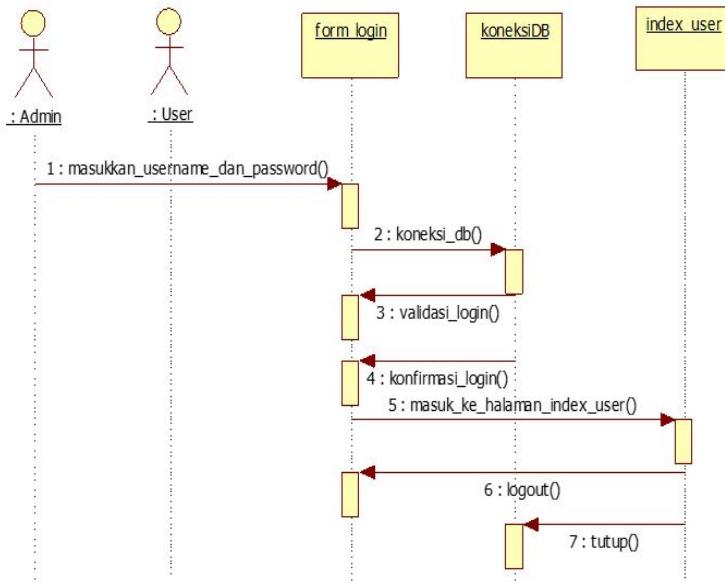
4.4.7 Class Diagram

Class diagram adalah diagram yang menggambarkan struktur sistem dan menunjukkan *class-class* yang ada dari sebuah sistem. Berikut adalah *class diagram* nya



Gambar 4.113 *Class Diagram*

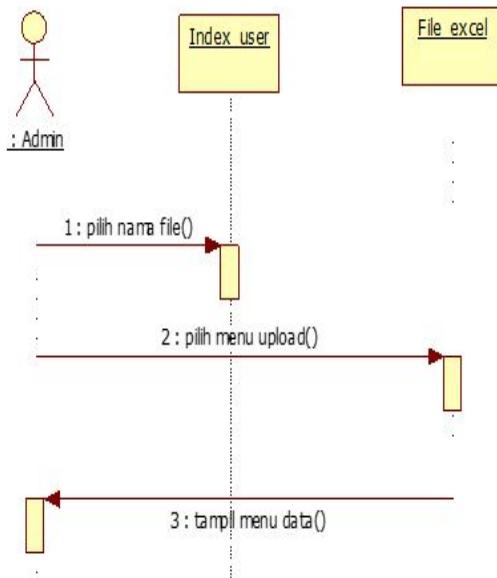
4.4.8 Sequence Diagram Login



Gambar 4.114 Sequence Diagram Login

Pada *Sequence diagram* ini menjelaskan proses aktor untuk melakukan *login*. Aktor mulai menjalankan aplikasi dan kemudian akan tampil *form login*. Kemudian aktor akan menginputkan Username dan password pada interface. Apabila *Username* dan *password* sesuai, maka aktor akan masuk ke halaman utama, apabila *Username* dan *password* tidak sesuai, maka aktor akan kembali ke halaman *login*.

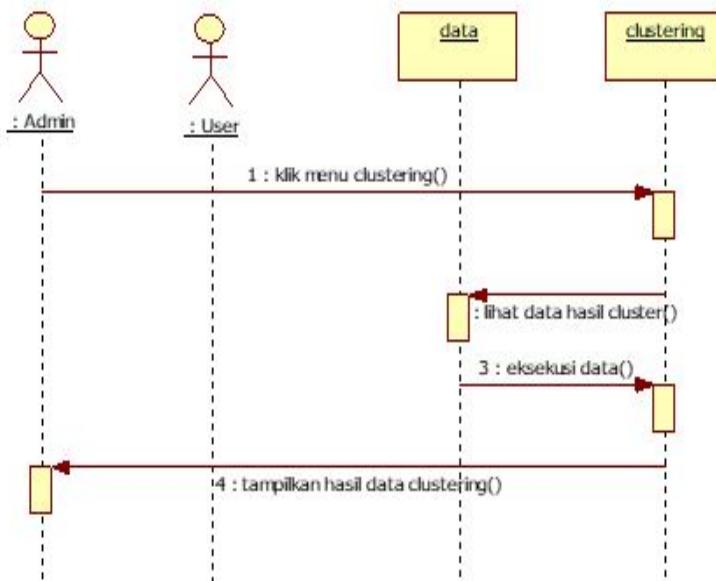
4.4.9 Sequence Diagram Upload File



Gambar 4.115 Sequence Diagram Upload File

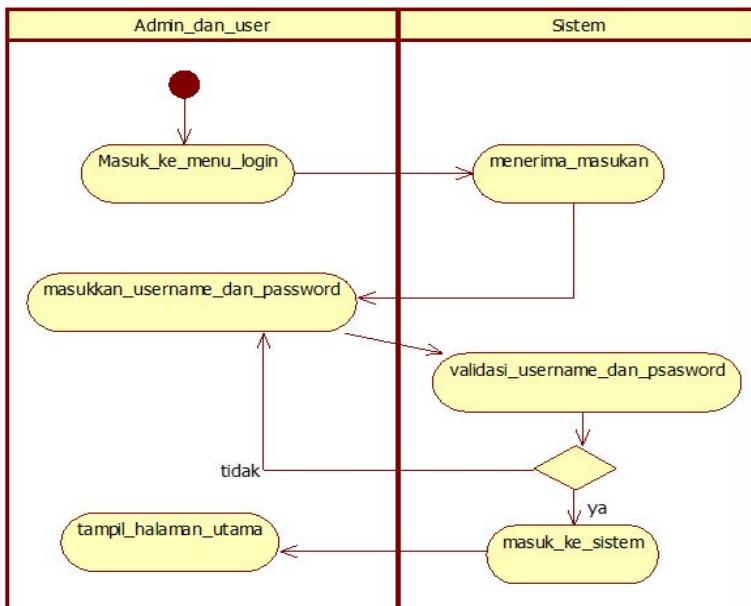
Sequence diagram ini menjelaskan tentang *sequence diagram upload file* yang dilakukan oleh admin. Admin akan memilih *file* yang akan dilakukan proses *clustering*, lalu kemudian admin akan memilih menu *upload*, dan kemudian sistem akan menampilkan data hasil *clustering*.

4.4.10 Sequence diagram Lihat Hasil Clustering



Gambar 4.116 Sequence Diagram Melihat Hasil Clustering

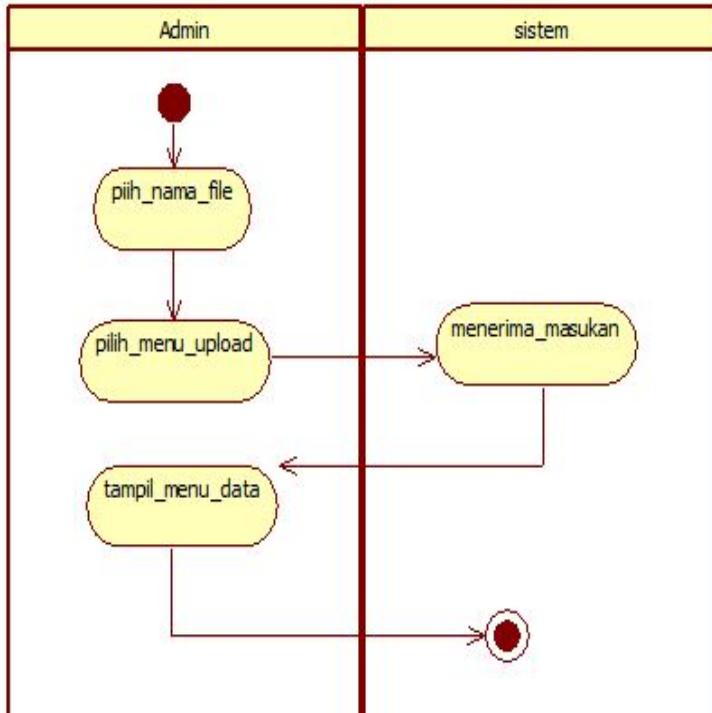
4.4.11 Activity diagram Login



Gambar 4.117 Activity Diagram Login

Pada *Activity diagram* ini menjelaskan proses *login*. Actor mulai menjalankan aplikasi dan akan tampil *form login*. Kemudian aktor memasukkan *Username* dan *password*, pada *interface login*, data yang di *input* akan di *cek* di tabel *User*. Jika sesuai maka akan mendapatkan tampilan *form utama pengguna*

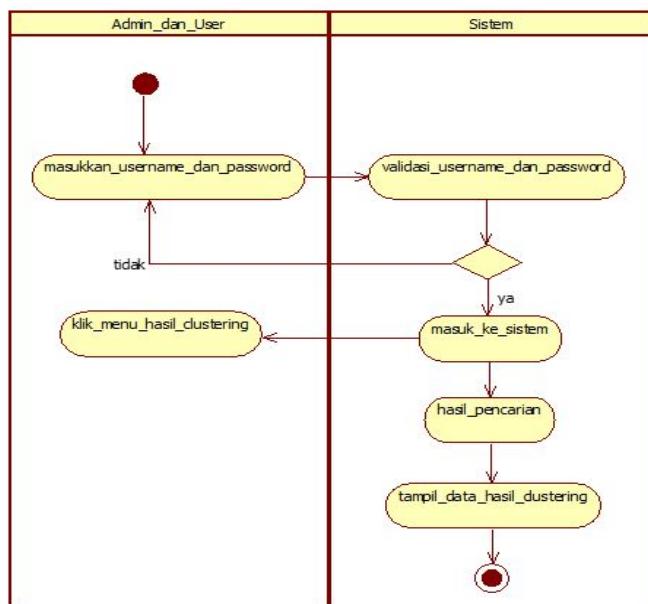
4.4.12 Activity diagram Upload File



Gambar 4.118 Activity Diagram Upload File

Pada *activity diagram* ini menjelaskan proses *upload file* yang dilakukan oleh admin. Pertama, admin akan memilih nama atau data *file* yang akan diolah. Kemudian admin memilih *button upload*. Kemudian sistem akan menerima masukan dan akan diproses dengan menggunakan metode *K-Means Clustering*. Setelah dilakukan pemrosesan data, maka akan tampil hasil data *clustering*.

4.4.13 Activity diagram Lihat Hasil Clustering



Gambar 4.119 Activity Diagram Lihat Hasil Cluster

Pada *activity diagram* ini menjelaskan proses lihat hasil *clustering*. Pertama, aktor memasukkan *Username* dan *password* pada menu *login*, kemudian sistem akan validasi *Username* dan *password* yang diinputkan. Apabila *Username* dan *password* dimasukkan benar, maka aktor akan masuk ke halaman utama, apabila tidak, maka aktor akan kembali ke halaman *login*. Kemudian setelah masuk ke halaman utama, aktor akan klik hasil *clustering*, dan sistem akan menampilkan hasil *clustering*

4.5 Perancangan Antarmuka

4.5.1 Perancangan Antarmuka Halaman Login

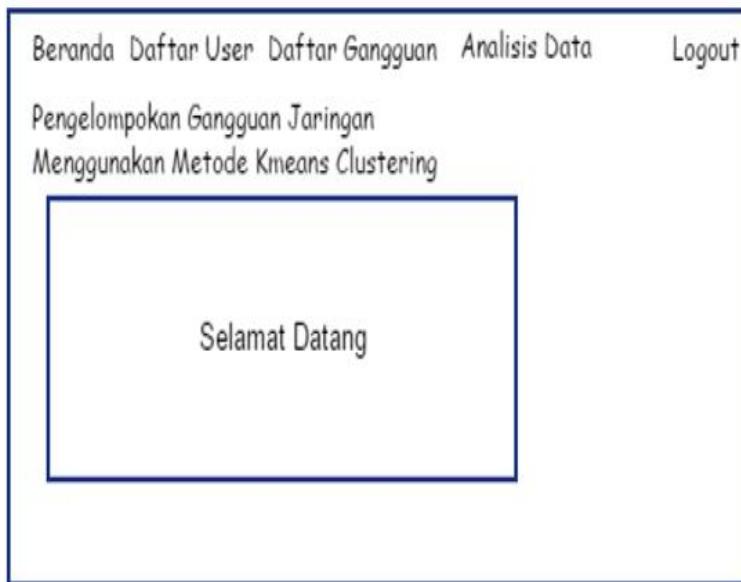


Gambar 4.120 Halaman Login

Keterangan:

Userinterface diatas menunjukkan bahwa untuk login ke halaman utama aplikasi, User harus memasukkan *Username* dan *password*

4.5.2 Perancangan Antarmuka Halaman Beranda

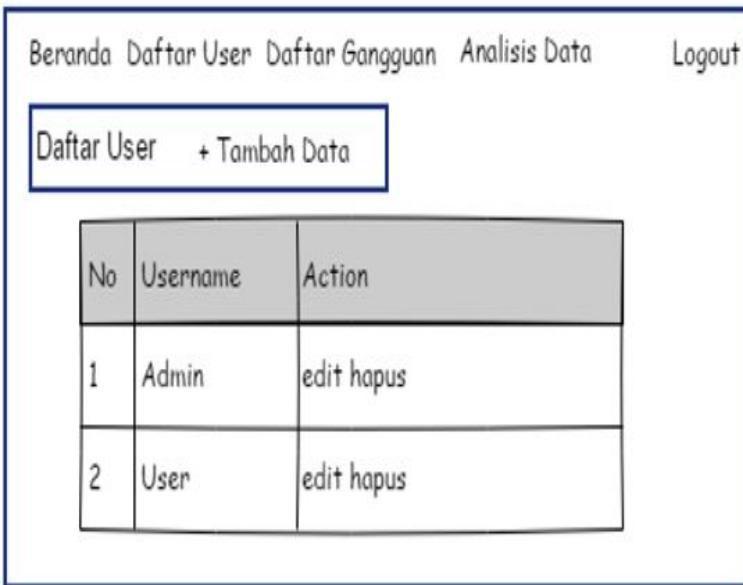


Gambar 4.121 Halaman Beranda

Keterangan:

User interface diatas menunjukkan halaman beranda yang terdiri atas 4 menu yang akan dibuka sesuai kebutuhannya, yaitu menu beranda, daftar user, daftar gangguan, analisis data.

4.5.3 Perancangan Antarmuka Halaman Daftar User

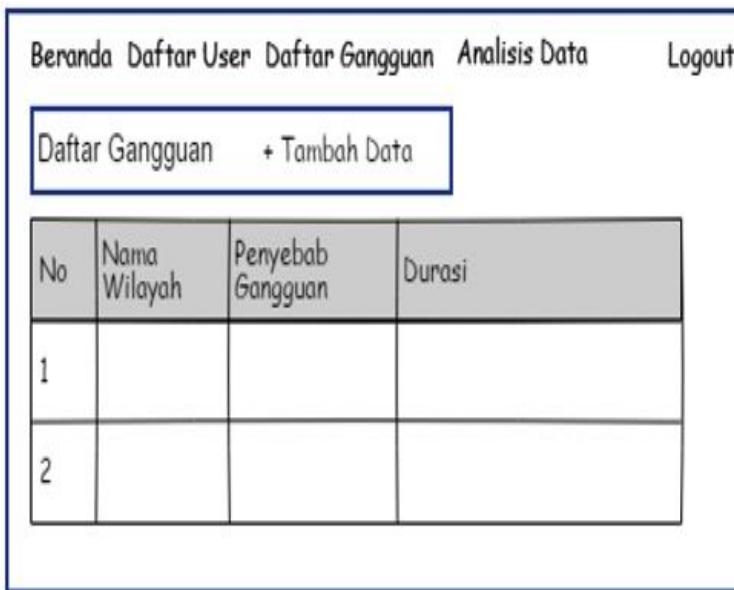


Gambar 4.122 Halaman Daftar User

Keterangan:

User interface diatas menunjukkan halaman *user* yang digunakan untuk mengelola *user* yang dapat mengakses aplikasi.

4.5.4 Perancangan Antarmuka Halaman Daftar Gangguan

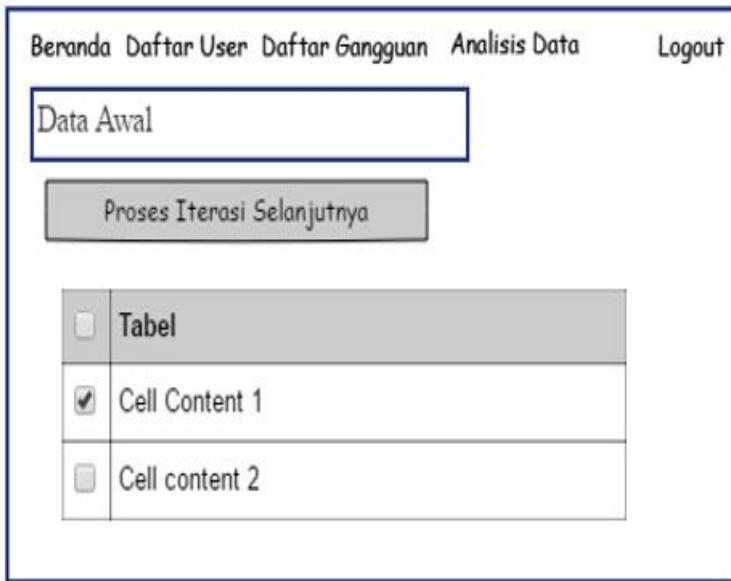


Gambar 4.123 Halaman Gangguan

Keterangan:

User interface diatas menunjukkan halaman daftar gangguan yang dimaksudkan adalah wilayah, penyebab gangguan, durasi. Terdapat tambah data untuk melakukan proses upload untuk memasukkan data excel kemudian dilakukan klasterisasi.

4.5.5 Perancangan Antarmuka Halaman Iterasi Data



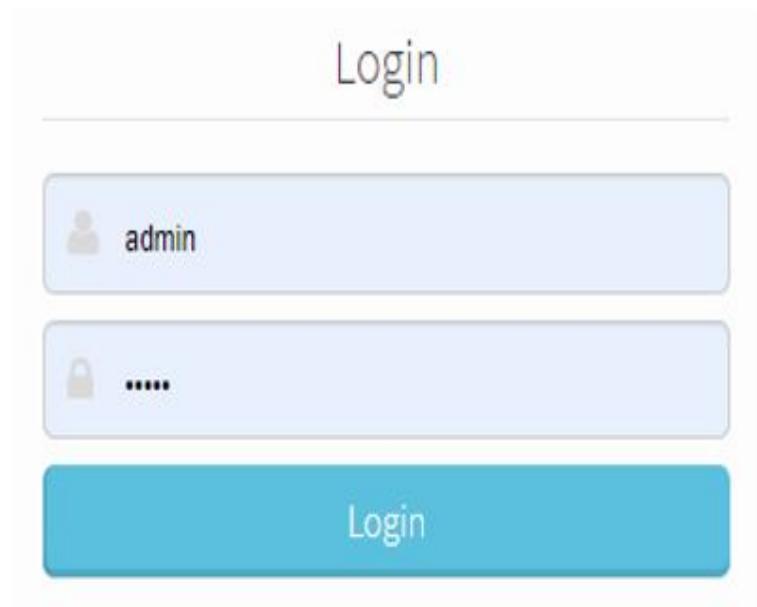
Gambar 4.124 Halaman Iterasi Data

Keterangan:

Userinterface diatas menunjukkan halaman iterasi data dari data gangguan yang telah di *upload* pada halaman daftar gangguan.

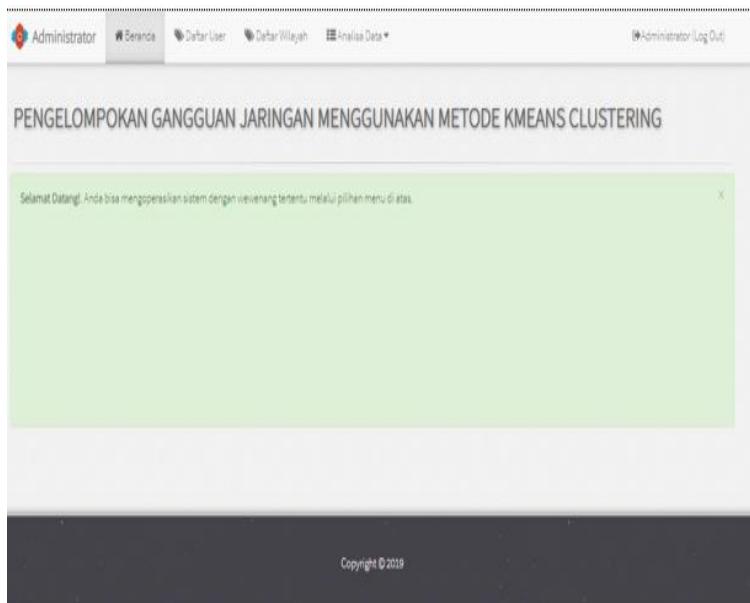
4.6 Interface

4.6.1 User Interface Login



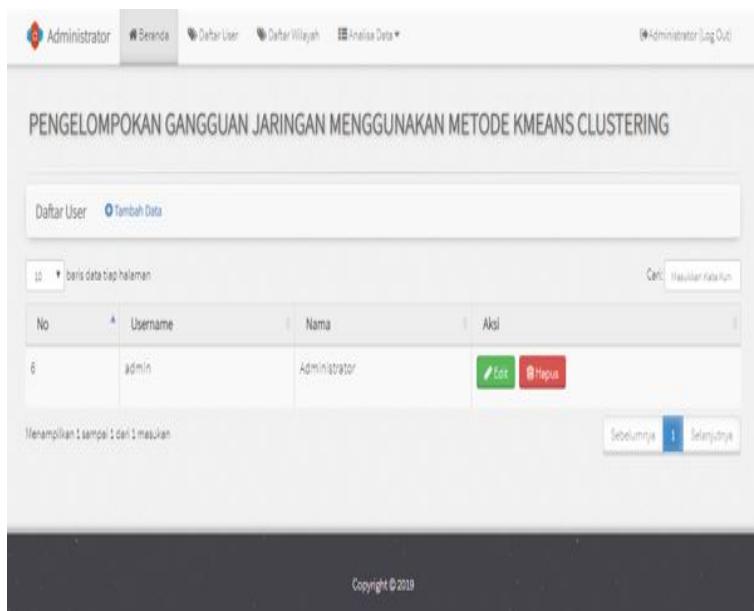
Gambar 4.125 Halaman Login

4.6.2 User Interface Beranda



Gambar 4.126 Halaman Beranda

4.6.3 User Interface Daftar User



Gambar 4.127 Halaman Daftar User

4.6.4 User Interface Jenis Gangguan

No	Nama Wilayah	Penyebab Gangguan	Inisialisasi Gangguan	Durasi	Inisialisasi Durasi
1501	BANDUNG BARAT	METRO	1	358	1
1502	TASIKMALAYA	ONT	4	443	1
1503	KARAWANG	ASSESORIS	6	158	2
1504	CIREBON	LOGIC	3	97	2
1505	CIREBON	ONT	4	143	2
1506	BANDUNG BARAT	ASSESORIS	6	110	2
1507	TASIKMALAYA	LOGIC	3	51	3
1508	TASIKMALAYA	METRO	1	58	3
1509	BANDUNG BARAT	ASSESORIS	6	142	2
1510	BANDUNG	PO	5	529	1

Menampilkan 1 sampai 10 dari 500 masukan.

Selanjutnya [1](#) [2](#) [3](#) [4](#) [5](#) ... [50](#) Selanjutnya

Gambar 4.128 Halaman Jenis Gangguan

4.6.5 User Interface Iterasi Kmeans Clustering

No	Wilayah	Nama Wilayah	Instansi Wilayah	Periyed Gangguan	Instansi Gangguan	Durasi (Menit)	Instansi Durasi	Centroid 1			Centroid 2			Centroid 3		
								2	3	2	4	7	1	3	3	3
1	BANDUNG BARAT	1	METRO	1	358	1		3.7416873867739	8.0827625802992	3.4641016181878	0	0	1			
2	TASIKMALAYA	1	ONT	4	445	1		1.4142138623731	3.608881275464	2.4494897427832	1	0	0			
3	KARAWANG	2	ASSESORIS	8	158	2		3.805551275464	1.4142138623731	3.3186247903854	0	1	0			
4	CIREBON	2	LOGIC	3	97	2		1	5.0990195138928	2.2380679774998	1	0	0			
5	CIREBON	2	ONT	4	145	2		1.4142138623731	4.3888899435407	2.4494897427832	1	0	0			
6	BANDUNG BARAT	2	ASSESORIS	8	110	2		4.2428406871193	1.7320580756889	3.7416873867739	0	1	0			
7	TASIKMALAYA	3	LOGIC	3	81	3		1	4.0809794855664	1	1	0	1			
8	TASIKMALAYA	3	PO	7	50	3		4.1231058258177	2.8284271247462	4.1231058258177	0	1	0			
9	KARAWANG	2	ASSESORIS	8	142	2		3.805551275464	1.4142138623731	3.3186247903854	0	1	0			
10	BANDUNG	1	MODUL	5	829	1		4.9826759849558	2.8284271247462	4.1231058258177	0	1	0			
11	BANDUNG	1	MODUL	5	278	1		4.5828758949558	2.8284271247462	4.1231058258177	0	1	0			
12	SUKABUMI	1	METRO	1	245	1		2.4494897427832	8.0827625802992	2.8284271247462	1	0	0			

Gambar 4.129 Halaman Iterasi Kmeans Clustering

4.7 Pengujian Blackbox

Blackbox Testing adalah pengujian yang hanya dilakukan dengan cara mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Jadi dianalogikan seperti kita melihat kotak hitam. Kita hanya dapat melihat penampilan luarnya saja, tanpa ada yang tau ada apa di balik bungkus hitam. Sama seperti pengujian *Blackbox*. Mengevaluasi hanya dari tampilan luarnya saja atau *userinterface*.

Kelas Uji	Butir Uji	Identifikasi Uji	Tingkat Pengujian	Jenis Pengujian	Jadwal Pengujian
Login	Berhasil masuk ke halaman utama	UC-01	Berhasil	Blackbox	13 Desember 2019
Kelola User	Dapat menampilkan, menambahkan, mengedit, dan menghapus pengguna	UC-02	Berhasil	Blackbox	13 Desember 2019
Perhitungan K-Means	Dapat mengupload dan menampilkan hasil perhitungan	UC-03	Berhasil	Blackbox	13 Desember 2019

Gambar 4.130 Pengujian Blackbox

BAB 5

PENGKAJIAN DAN EVALUASI

5.1 Penerapan Algoritma K-Means Clustering

Secara umum, system yang akan dibangun dalam penelitian ini adalah sebuah sistem dengan fungsi utama untuk melakukan clustering gangguan. Data yang digunakan untuk penelitian merupakan data yang diperoleh dari PT. Telkom Indonesia divisi *OLO Fulfillment Assurance* yakni gangguan jaringan Telkomsel dengan atribut: wilayah, jenis gangguan jaringan, dan durasi.

5.1.1 Transformasi Data

Agar data dapat diolah dengan menggunakan metode *K-Means Clustering*, maka data yang berjenis data nominal seperti wilayah, jenis gangguan jaringan, dan durasi harus diinisialisasikan terlebih dahulu dalam bentuk angka.

Inisialisasi Wilayah

Wilayah	Inisial
Bandung	6
Bandung Barat	5
Karawang	4
Sukabumi	3
Tasikmalaya	2
Cirebon	1

Gambar 5.1 Inisialisasi Wilayah

Inisialisasi Gangguan

Jenis Gangguan	Inisial	Jenis Gangguan	Inisial
FO	7	Logic	3
Assesoris	6	Cada	2
Modul	5	Metro	1
Ont	4		

Gambar 5.2 Inisialisasi Gangguan

Inisialisasi Durasi

Durasi	Bobot
<1 Jam	3
1-4 Jam	2
>4 Jam	1

Gambar 5.3 Inisialisasi Durasi

5.1.2 Pengolahan Data

Setelah semua data ditransformasi ke dalam bentuk angka, maka data-data tersebut telah dapat dikelompokkan dengan menggunakan algoritma *K-Means Clustering*. Untuk dapat melakukan pengelompokan data-data tersebut menjadi beberapa *cluster* perlu dilakukan beberapa langkah, yaitu:

1. Tentukan jumlah *cluster* yang diinginkan. Dalam penelitian ini data yang ada akan dikelompokkan dalam 3 cluster atau $K=3$. Penentuan K tersebut berdasarkan metode *elbow*.
2. Tentukan titik pusat awal dari setiap cluster. Dalam penelitian ini titik pusat awal ditentukan secara *random* dan didapat titik pusat dari setiap *cluster*
3. Setelah diketahui nilai k dan pusat *cluster* awal selanjutnya mengukur jarak antara pusat *cluster* menggunakan *Euclidian distance*
4. Menentukan titik pusat cluster baru

Hitung kembali setiap data dengan titik pusat cluster baru. Jika pusat cluster tidak berubah lagi maka proses cluster selesai.

5.1.3 Hasil Clustering

Hasil *Clustering* yang dimaksudkan adalah data yang diolah sesuai dengan data yang diberikan oleh perusahaan sejumlah 500 data. Berdasarkan data yang sudah ada, hasil perhitungan menggunakan metode *k-means clustering* bahwa perhitungan tersebut menggunakan atribut wilayah, jenis gangguan, dan durasi menghasilkan *cluster* yang berbeda. Hasil *cluster* tersebut adalah sebagai berikut

Cluster 1 terdiri dari 176 data. *Cluster 2* terdiri dari 173 data. *Cluster 3* terdiri dari 151 data. Terdapat 176 data pada *Cluster 1*, *Cluster* tersebut termasuk dalam kategori wilayah yang dekat dengan pusat yaitu Bandung dengan jumlah data sebanyak

68 dan Bandung Barat sebanyak 67 kemudian gangguan didominasi oleh gangguan FO dengan penanganan yang cukup baik, yaitu selama 1-4 jam.

Cluster 2 termasuk ke dalam kategori dengan wilayah yang cukup jauh dari pusat(bandung) yaitu wilayah Tasikmalaya dengan jumlah data sebanyak 81 dengan gangguan assesoris sebanyak 54 dan penanganan yang hampir seimbang antara durasi penanganan yang sedang dan durasi penanganan yang lambat.

Cluster 3 terdiri dari 151 data. Pada cluster ini tasikmalaya mendominasi cluster tersebut dengan gangguan terbanyak cada yaitu 50 data. Sebagian besar penanganan tersebut dilakukan lebih dari 1jam. Maka cluster 3 termasuk ke dalam kategori yang wilayah yang lumayan dekat dari pusat dengan gangguan Cada dengan durasi penanganan didominasi lebih dari 1jam

DAFTAR PUSTAKA

1. P. D. Astuti, “Perancangan sistem informasi penjualan obat pada apotek jati farma arjosari,” in *Seruni-Seminar Riset Unggulan Nasional Informatika dan Komputer*, vol. 2, no. 1, 2013.
2. D. Rahmadiansyah, D. Irwan, D. Sekolah, and T. Teknik, “Implementasi metode model view controller menggunakan framework code igniter dalam pengembangan aplikasi manajemen depo petikemas pada unit usaha belawan logistics center,” in *Seminar Nasional Teknologi Informasi dan Komunikasi (SNASTIKOM) 2012*, 2012, pp. 1–11.
3. I. A. Susila and R. Taufiq, “Penerapan metode analytical heirarchy process (ahp) dalam sistem pendukung keputusan (spk) pemensiunan pada badan kepegawaian dan pengembangan sumber daya manusia kota tangerang,” 2018.
4. D. D. Dvorski, “Installing, configuring, and developing with xampp,” *Skills Canada*, 2007.
5. A. Magno, *Mobile-first Bootstrap*. Packt Publishing Ltd, 2013.
6. A. Pareek, M. Lakshminarayanan, A. Dubey, and S. Corbin, “Mysql database heterogeneous log based replication,” Aug. 13 2013, uS Patent 8,510,270.

