

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Pembuatan aplikasi atau program tentang sistem untuk memprediksi prestasi akademik sudah banyak dibuat, namun program aplikasinya berbeda-beda. Adapun aplikasi sistem untuk memprediksi prestasi akademik yang berkaitan dengan menggunakan metode *K-Nearest Neighbor* yang pernah dibuat adalah sebagai berikut.

Mustakim, M., & Oktaviani, G. (2016) menghasilkan sistem prediksi predikat prestasi akademik pada mahasiswa. Adapun kesimpulan yang dihasilkan dari penelitian tersebut adalah perhitungan algoritma K-NN yang diterapkan dalam memprediksi predikat prestasi mahasiswa mampu menghasilkan akurasi dengan nilai 82%. Pengujian algoritma ini dilakukan menggunakan perhitungan confusion matriks yaitu membandingkan predikat pada semester sebelumnya dengan predikat hasil prediksi. Proses prediksi predikat prestasi Mahasiswa dilakukan dengan membangun sebuah *Early Warning System (EWS)* berdasarkan algoritma K-NN. Output yang dihasilkan dari penelitian ini adalah daftar predikat prestasi akademik dari masing-masing mahasiswa.

Niswatin, R. K. (2016) menghasilkan sebuah sistem pendukung keputusan untuk membantu proses penempatan jurusan mahasiswa baru menggunakan metode *k-nearest neighbor*. Rekomendasi jurusan diperoleh berdasarkan data training yang diambil pada mahasiswa jurusan teknik informatika dan sistem informasi angkatan 2012, kemudian diproses menggunakan metode *k-nearest neighbor* sehingga menghasilkan rekomendasi jurusan bagi mahasiswa data testing jurusan teknik informatika dan sistem informasi angkatan 2013. Hasil penelitian ini menunjukkan bahwa *metode k-nearest neighbor* merupakan metode yang cukup baik dan sesuai

digunakan untuk menyelesaikan permasalahan klasifikasi. Syarat utama penggunaan metode *k-nearest neighbor* untuk menyelesaikan permasalahan klasifikasi adalah tersedianya data *training* yang baik dan akurat, karena pada metode *k-nearest neighbor* hasil klasifikasi diperoleh dengan menghitung kedekatan antara permasalahan baru (data *testing*) dengan permasalahan lama (data *training*) berdasarkan pada kecocokan bobot / nilai dari fitur – fitur yang telah ditentukan.

Ginting, S. L. B., Zarman, W., & Darmawan, A. (2014) penelitian ini menghasilkan sistem untuk memprediksi masa studi mahasiswa menggunakan algoritma *k-nearest neighborhood*. Adapun beberapa kesimpulan yang dihasilkan, yaitu:

1. Data training dengan jumlah 30 data digunakan untuk menguji data testing berjumlah 30 data, didapatkan nilai *k* yang terbaik untuk memprediksi masa studi mahasiswa yaitu sebagai berikut:

- a. Untuk dua semester yaitu nilai *k* yang terbaik untuk digunakan memprediksi studi mahasiswa adalah nilai $k=10$ dengan tingkat keberhasilan 81.66%.
- b. Untuk empat semester yaitu nilai *k* yang terbaik untuk digunakan memprediksi studi mahasiswa adalah nilai $k=10$ dengan tingkat keberhasilan 76.66%. Untuk enam semester yaitu nilai $k=10$ merupakan nilai *k* yang terbaik untuk digunakan memprediksi masa studi mahasiswa dengan tingkat keberhasilan 81.66%.

2. Data training dengan jumlah 60 data digunakan untuk menguji data testing berjumlah 61 data, diperoleh nilai *k* yang terbaik untuk memprediksi masa studi mahasiswa yaitu sebagai berikut:

- a. Untuk dua semester yaitu nilai *k* yang terbaik untuk memprediksi masa studi mahasiswa adalah nilai $k=10$ dengan tingkat keberhasilan 80%.
- b. Untuk empat semester, nilai *k* yang terbaik untuk memprediksi masa studi mahasiswa adalah nilai $k=30$ dengan tingkat keberhasilan 78.33%.
- c. Untuk enam semester, nilai *k* yang terbaik untuk memprediksi masa studi mahasiswa adalah nilai $k=20$ dan $k=30$ dengan tingkat keberhasilan 85%.

Banjarsari, M. A., Budiman, I., & Farmadi, A. (2016) pada penelitian ini menghasilkan sistem untuk prediksi kelulusan tepat waktu mahasiswa program studi ilmu komputer Fmipa Unlam Berdasarkan IP Sampai Dengan Semester 4. Kesimpulan yang diperoleh dari penelitian ini adalah :

1. Nilai k -Optimal pada algoritma KNN untuk prediksi kelulusan tepat waktu mahasiswa berdasarkan IP sampai dengan semester 4 adalah $k=5$.

2. Dari proses *k-Fold Cross Validation* didapatkan tingkat akurasi untuk $k=5$ pada algoritma kNN untuk prediksi kelulusan tepat waktu mahasiswa berdasarkan IP sampai dengan semester 4 adalah sebesar 80,00%.

Mustafa, M. S., & Simpen, I. W. (2014) judul penelitian ini adalah Perancangan Aplikasi Prediksi Kelulusan Tepat Waktu Bagi Mahasiswa Baru Dengan Teknik Data Mining (Studi Kasus: Data Akademik Mahasiswa STMIK Dipanegara Makassar). Kesimpulan yang dihasilkan ialah kita dapat mengetahui hubungan kedekatan antara kasus yang baru dengan kasus yang telah ada dalam suatu gudang data (*data warehouse*) sehingga dapat menjadi acuan untuk memprediksi kelulusan seorang mahasiswa baru apakah dapat menyelesaikan kuliahnya dengan tepat waktu atau tidak berdasarkan kedekatan data yang sudah ada. Dari hasil pengujian dengan menerapkan algoritma KNN dan menggunakan data sampel alumni tahun wisuda 2004 s.d. 2010 untuk kasus lama dan data alumni tahun wisuda 2011 untuk kasus baru diperoleh tingkat akurasi sebesar 83,36%.

Pradnyana, G. A., & Permana, A. A. J. (2017) judul penelitian ini adalah Perancangan Sistem Pembagian Kelas Kuliah Mahasiswa dengan Kombinasi Metode K-Means dan K-Nearest Neighbors. Kesimpulan dalam penelitian ini adalah Penelitian ini telah berhasil merancang sebuah metode baru untuk pembagian kelas kuliah mahasiswa. Penelitian ini menggunakan studi kasus di Jurusan Pendidikan Teknik Informatika Universitas Pendidikan Ganesha. Tahap awal dari metode yang dikembangkan adalah memproses data nilai kuliah mahasiswa sebelumnya yang telah mengambil mata kuliah, untuk dijadikan data training dalam proses prediksi dengan metode KNN. Hasil dari proses ini adalah prediksi kelulusan masing-masing mahasiswa yang akan mengambil mata kuliah atau yang akan dikelompokkan ke dalam beberapa kelas. Nilai prediksi kelulusan mahasiswa ini akan digabungkan dengan nilai tugas, ujian tengah semester (UTS), dan ujian akhir semester (UAS) pada mata kuliah prasyarat serta IPK semester mahasiswa untuk dijadikan fitur dalam proses pengelompokkan (*clustering*) oleh metode K-means.

Risman, H., Nugroho, D., & Utami, Y. R. W. (2015). Judul penelitian ini adalah Penerapan Metode *K-Nearest Neighbor* Pada Aplikasi Penentu Penerima Beasiswa Mahasiswa Di STMIK Sinar Nusantara Surakarta. Adapun kesimpulan yang bisa diambil sebagai berikut :

1. Metode *K-Nearest Neighbor (K-NN)* merupakan metode klasifikasi yang dapat digunakan untuk memberikan keputusan alternatif dalam menentukan calon penerima beasiswa berdasarkan data-data yang telah diperoleh (data sampel).

2. Kinerja sistem berdasarkan data sampel yang digunakan mencapai 90,90% dan termasuk hasil yang baik mengingat bahwa metode *K-Nearest Neighbor* tidak menggunakan parameter yang telah ditentukan untuk menentukan keputusan melainkan menggunakan data sampel yang nilainya bervariasi.

Rohman, A. (2015) menghasilkan sistem untuk prediksi kelulusan mahasiswa. Adapun kesimpulan dalam penelitian ini adalah dalam penelitian ini dilakukan pengujian model dengan menggunakan algoritma *K-Nearest Neighbor (K-NN)* dengan menggunakan data kelulusan mahasiswa yang tepat dan terlambat. Model yang dihasilkan diuji untuk mendapatkan nilai *accuracy* dan AUC dari algoritma klasifikasi data mining sehingga didapat pengujian data mahasiswa dengan klatering data $k=1$ dengan menggunakan *K-Nearest Neighbor (K-NN)* didapat nilai *accuracy* adalah 82,25% dan nilai AUC adalah 0.500, dengan cluster data $k=2$ *accuracy* adalah 79,45% dan nilai AUC adalah 0.826, dengan cluster data $k=3$ *accuracy* adalah 83,95% dan nilai AUC adalah 0.853, dengan cluster data $k=4$ *accuracy* adalah 82,62% dan nilai AUC adalah 0.874, dengan cluster data $k=5$ *accuracy* adalah 85,15% dan nilai AUC adalah 0.888.

Sumarlin, S. (2015) Judul penelitian ini adalah Implementasi Algoritma *K-Nearest Neighbor* Sebagai Pendukung Keputusan Klasifikasi Penerima Beasiswa PPA dan BBM. Kesimpulan dari penelitian disini ialah mengimplementasikan algoritma *k-nearest neighbor* pada data penerimaan beasiswa PPA dan BBM. Untuk mencari data yang berkualitas, dilakukan preprosesing sebelum diterapkan kedalam algoritma. Kedekatan antara kasus baru dengan kasus lama dilakukan untuk menentukan pada kelas mana kasus baru akan diklasifikasikan. Dengan membangun sistem pendukung keputusan untuk mengklasifikasikan beasiswa PPA dan BBM yang ditentukan atau dianalisis dengan menggunakan algoritma *k-nearest neighbor*. Ada enam variable yang

digunakan yaitu indeks prestasi kumulatif, semester, piagam penghargaan, tanggungan orang tua, pendapatan orang tua dan tagihan listrik.

Hasil uji verifikasi yang ada menunjukkan bahwa sistem pendukung keputusan yang dibuat dengan menggunakan analisis algoritma *k-nearest neighbor* menghasilkan keluaran yang sama dengan perhitungan manual yang dilakukan dengan microsoft excel, dimana keluaran sistem berupa nilai terendah yang dijadikan kasus terdekat untuk mengklasifikasi kasus baru sama dengan hasil perhitungan microsoft excel. Untuk mengukur kinerja algoritma *k-nearest neighbor* digunakan metode *Cross Validation*, *Confusion Matrix* dan Kurva ROC, dalam penelitian ini menggunakan *10-fold cross validation* untuk klasifikasi beasiswa PPA dan BBM. Dari 227 record dataset beasiswa PPA diperoleh tingkat *accuracy* mencapai 77,96% dan termasuk klasifikasi sangat baik karena memiliki nilai AUC berada diantara 0,90-1,00, yaitu 0,925, sedangkan untuk beasiswa BBM digunakan 183 record dataset dan diperoleh tingkat *accuracy* mencapai 97,28% dengan nilai AUC 0,937, termasuk dalam klasifikasi sangat baik. Akurasi. Untuk gabungan beasiswa PPA dan BBM diperoleh *accuracy* mencapai 85,56% dan memiliki nilai AUC 0,958 juga termasuk klasifikasi sangat baik.

Susanto, H., & Sudiyatno, S. (2014) menghasilkan sistem prediksi prestasi akademik pada siswa berdasarkan sosial ekonomi, motivasi, kedisiplinan dan prestasi masa lalu, analisis data menggunakan *decision tree* data mining untuk memprediksi prestasi belajar siswa berdasarkan status sosial ekonomi orang tua, motivasi, kedisiplinan siswa dan prestasi masa lalu menggunakan metode data mining diperoleh hasil sebagai berikut :

1. Variabel motivasi adalah variabel yang menentukan potensi seorang siswa berhasil atau tidak prestasi belajarnya di waktu yang akan datang. Hal ini dibuktikan dengan adanya variabel motivasi yang menjadi *root node* dalam *decision tree* yang terbentuk. Variabel prestasi masa lalu merupakan variabel kedua yang penting dalam keberhasilan siswa menempuh studinya. Hal ini menunjukkan bahwa aspek *knowledge* atau kecerdasan siswa sangat berpengaruh terhadap keberhasilan belajarnya. Sebaliknya, walaupun siswa terprediksi secara *knowledge* kurang tetapi dengan motivasi tinggi tetap bisa berprestasi minimal pada kategori B atau C. Rata-rata keberhasilan algoritma J48 dalam melakukan klasifikasi data mencapai akurasi di 95,7%. Hal ini menunjukkan bahwa algoritma ini memiliki performa yang handal dalam melakukan klasifikasi.

2. Hasil akurasi klasifikasi menggunakan metode CHAID dengan data yang sama memperoleh hasil rata-rata prediksi 82,1. Hal ini menunjukkan bahwa berdasarkan hasil klasifikasi menggunakan metode CHAID tersebut memiliki tingkat akurasi yang tergolong baik, yaitu lebih dari 80%; (3) Berdasarkan tabel di atas angka R Square adalah 0,906 untuk predictor Motivasi dan Prestasi_ml. Hasil tersebut adalah hasil kuadrat korelasi. *Standar Error of the Estimate* adalah 0,30477, perhatikan pada analisis deskriptif statistik bahwa standar deviasi nilai uts adalah 0,99246 yang jauh lebih besar dari dari standar *error*, oleh karena lebih besar daripada standar deviasi nilai uts maka model regresi ini bagus dalam bertindak sebagai predictor nilai uts.

Berdasarkan ketiga hasil penelitian diatas dapat diketahui bahwa akurasi prediksi prestasi belajar menggunakan metode *J48* adalah sebesar 95,7%, sedangkan metode CHAID mempunyai akurasi prediksi sebesar 82,1% dan analisis regresi ganda mempunyai tingkat signifikansi 90,6%. Berdasarkan data tersebut dapat disimpulkan bahwa penggunaan metode data mining dengan algoritma *decision tree* (*J48*) memiliki tingkat akurasi yang lebih baik dari 2 (dua) metode lain yaitu CHAID dan regresi ganda.

2.2 K-NEAREST NEIGHBOR (K-NN)

Algoritma *K-Nearest Neighbor* (K-NN) adalah sebuah metode untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. *K-Nearest Neighbor* berdasarkan konsep '*learning by analogy*'. Data *learning* dideskripsikan dengan atribut numerik *n*-dimensi. Tiap data *learning* merepresentasikan sebuah titik, yang ditandai dengan *c*, dalam ruang *n*-dimensi. Jika sebuah data *query* yang labelnya tidak diketahui diinputkan, maka *K-Nearest Neighbor* akan mencari *k* buah data *learning* yang jaraknya paling dekat dengan data *query* dalam ruang *n*-dimensi. Jarak antara data *query* dengan data *learning* dihitung dengan cara mengukur jarak antara titik yang merepresentasikan data *query* dengan semua titik yang merepresentasikan data *learning* dengan rumus *Euclidean Distance*. (Hamitha dkk., 2016)

Pada fase *training*, algoritma ini hanya melakukan penyimpanan vektor-vektor fitur dan klasifikasi data *training sample*. Pada fase klasifikasi, fitur-fitur yang sama dihitung untuk *testing data* (klasifikasinya belum diketahui). Jarak dari vektor yang baru ini terhadap seluruh vektor *training sample* dihitung, dan sejumlah *k* buah yang paling dekat diambil. Titik yang baru klasifikasinya diprediksikan termasuk pada klasifikasi terbanyak dari titik-titik tersebut. Nilai *k* yang terbaik untuk algoritma ini tergantung pada

data secara umumnya, nilai k yang tinggi akan mengurangi efek *noise* pada klasifikasi, tetapi membuat batasan antara setiap klasifikasi menjadi lebih kabur. Nilai k yang bagus dapat dipilih dengan optimasi parameter, misalnya dengan menggunakan *cross-validation*. Kasus khusus di mana klasifikasi diprediksikan berdasarkan data pembelajaran yang paling dekat (dengan kata lain, $k = 1$) disebut algoritma *nearest neighbor* (Sikki, 2012).

Ketepatan algoritma K-NN ini sangat dipengaruhi oleh ada atau tidaknya fitur-fitur yang tidak relevan, atau jika bobot fitur tersebut tidak setara dengan relevansinya terhadap klasifikasi. Riset terhadap algoritma ini sebagian besar membahas bagaimana memilih dan memberi bobot terhadap fitur, agar performa klasifikasi menjadi lebih baik.

2.2.1 Langkah-langkah Klasifikasi

Algoritma *K-Nearest Neighbor* menggunakan klasifikasi ketetanggaan (*neighbor*) sebagai nilai prediksi dari *query instance* yang baru. Algoritma ini sederhana, bekerja berdasarkan jarak terpendek dari *query instance* ke *training sample* untuk menentukan ketetanggaannya (Rizal, 2013). Langkah-langkah untuk menghitung metode *k-Nearest Neighbor* antara lain:

- a. Menentukan parameter k
- b. Menghitung jarak antara data yang akan dievaluasi dengan semua pelatihan
- c. Mengurutkan jarak yang terbentuk
- d. Menentukan jarak terdekat sampai urutan k
- e. Memasangkan kelas yang bersesuaian
- f. Mencari jumlah kelas dari tetangga yang terdekat dan tetapkan kelas tersebut sebagai kelas data yang akan dievaluasi.

$$d_i = \sqrt{\sum_{i=1}^p (x_{2i} - x_{1i})^2} \quad (1)$$

Dengan :

x_1 = sampel data

x_2 = data uji

i = variabel data

d = jarak

p = dimensi data

Fungsi jarak hanya digunakan pada variabel kontinyu, Untuk variabel kategori menggunakan *Hamming*. Diganti menggunakan standar penomoran variabel antara 0 dan 1, jika disubstitusikan antara angka dan variabel kategori pada dataset. Dengan rumusan sebagai berikut :

$$D_H = \sum_{i=1}^k |x_1 - y_1| \quad (2)$$

$$x = y \Rightarrow D = 0$$

$$x \neq y \Rightarrow D = 1$$

Pada fase training, algoritma ini hanya melakukan penyimpanan vektor-vektor dan klasifikasi data training sample. Pada fase klasifikasi, fitur-fitur yang sama dihitung untuk *testing* data (klasifikasinya belum diketahui). Jarak dari vektor yang baru ini terhadap seluruh vektor *training*, dan sejumlah k buah yang paling dekat diambil. Titik yang baru klasifikasinya diprediksikan termasuk pada klasifikasi terbanyak dari titik-titik tersebut (Husumardiana, 2015).

2.2.2 Menghitung Jarak *Euclidean*

Ada banyak cara untuk belajar mengukur jarak kedekatan antara data baru dengan data lama (*data training*), diantaranya *euclidean distance* dan *manhattan distance* (*city block distance*), yang paling sering digunakan adalah *euclian distance* (Bramer, 2007), yaitu :

$$d = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 + \dots + (a_n - b_n)^2} \quad (3)$$

Dimana $a = a_1, a_2, \dots, a_n$, dan $b = b_1, b_2, \dots, b_n$ mewakili n nilai atribut dari *record*. Untuk atribut dengan nilai kategori, pengukuran dengan *euclidean distance* tidak cocok. Sebagai penggantinya, digunakan fungsi sebagai berikut (Larose, 2006):

$$\text{Different } (a_i, b_i) \{0 \text{ jika } a_i = b_i\} \quad (4)$$

Dimana a_i dan b_i adalah nilai kategori. Jika nilai atribut antara dua *record* yang dibandingkan sama, maka nilai jaraknya 0, artinya mirip. Sebaliknya, jika berbeda maka

nilai kedekatannya 1, artinya tidak mirip sama sekali. Misal atribut warna dengan nilai merah dan merah, maka nilai kedekatannya 0, jika merah dan biru maka nilai kedekatannya 1.

2.2.3 Confusion matrix

Pengukuran terhadap kinerja suatu sistem klasifikasi merupakan hal yang penting. Kinerja sistem klasifikasi menggambarkan seberapa baik sistem dalam mengklasifikasikan data. Confusion matrix merupakan salah satu metode yang dapat digunakan untuk mengukur kinerja suatu metode klasifikasi. Pada dasarnya confusion matrix mengandung informasi yang membandingkan hasil klasifikasi yang dilakukan oleh sistem dengan hasil klasifikasi yang seharusnya (Prasetyo, 2012). Adapun perhitungan tingkat akurasi pada confusion matriks ialah sebagai berikut :

$$\text{Akurasi} = \frac{\text{Jumlah nilai benar}}{\text{Jumlah data keseluruhan}} \times 100\% \quad (5)$$

2.3 DATA MINING

Data mining adalah suatu istilah yang digunakan untuk menguraikan penemuan pengetahuan didalam *database*. *Data mining* adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan dan *mechine learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai database besar (Pane, 2013).

Penemuan pola-pola yang tidak diketahui sebelumnya, dimana data mining menyapu basis data, kemudian mengidentifikasi pola-pola yang sebelumnya tersembunyi dalam satu sapuan. Data mining berguna untuk membuat keputusan yang kritis, terutama dalam strategi (Moertini, 2002).

Data Mining mempunyai fungsi yang penting untuk membantu mendapatkan informasi yang berguna serta meningkatkan pengetahuan bagi pengguna. Pada dasarnya, data mining mempunyai empat fungsi dasar yaitu:

- a. Fungsi Prediksi (*prediction*). Proses untuk menemukan pola dari data dengan menggunakan beberapa variabel untuk memprediksikan variabel lain yang tidak diketahui jenis atau nilainya.
- b. Fungsi Deskripsi (*description*). Proses untuk menemukan suatu karakteristik penting dari data dalam suatu basis data.

- c. Fungsi Klasifikasi (*classification*). Klasifikasi merupakan suatu proses untuk menemukan model atau fungsi untuk menggambarkan class atau konsep dari suatu data. Proses yang digunakan untuk mendeskripsikan data yang penting serta dapat meramalkan kecenderungan data pada masa depan.
- d. Fungsi Asosiasi (*association*). Proses ini digunakan untuk menemukan suatu hubungan yang terdapat pada nilai atribut dari sekumpulan data.

2.4 PHP

PHP adalah sebuah kepanjangan dari *Hypertext Preprocessor*, PHP Atau *Hypertext Preprocessor* ialah sebuah bahasa pemrograman yang berupa kode atau script yang bisa ditambahkan ke dalam Bahasa Pemrograman HTML , PHP itu sendiri sering kali digunakan untuk hal merancang, membuat dan juga memprogram sebuah website. PHP juga sangat sering digunakan untuk membuat sebuah ataupun beberapa CMS, CMS ialah sebuah *software* atau perangkat lunak yang mempunyai kegunaan untuk memanipulasi semua atau beberapa isi dari sebuah halaman website (Oktavian, 2010).

2.5 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: *database management system*) atau DBMS yang *multithread*, *multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi GNU *General Public License* (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL (Andayati, 2010).

Relational Database Management System, (RDBMS) MySQL adalah *Relational Database Management System* (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (*General Public License*). Dimana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian database, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Keandalan suatu sistem database (DBMS) dapat diketahui dari cara kerja optimizer-nya dalam melakukan proses perintah-perintah SQL, yang dibuat oleh user maupun program-program aplikasinya. Sebagai database server, MySQL dapat dikatakan lebih unggul dibandingkan database

server lainnya dalam query data. Hal ini terbukti untuk query yang dilakukan oleh single user, kecepatan query MySQL bisa sepuluh kali lebih cepat dari PostgreSQL dan lima kali lebih cepat dibandingkan *Interbase* (Putri, 2012).

2.6 BOOTSTRAP

Bootstrap adalah sebuah *library framework* CSS yang dibuat khusus untuk bagian pengembangan *front-end website*. *Bootstrap* juga merupakan salah satu *framework HTML*, CSS dan javascript yang paling populer di kalangan *web developer* yang digunakan untuk mengembangkan sebuah website yang *responsive*. Sehingga halaman website nantinya dapat menyesuaikan sesuai dengan ukuran monitor device (desktop, tablet, ponsel) yang digunakan pengguna disaat mengakses website website dari *browser* (Muslikh, 2011).