

SKRIPSI

**IMPLEMENTASI *INTEGRATED DEVELOPMENT ENVIRONMENT (IDE)*
BERBASIS WEB UNTUK PEMROGRAMAN *JAVASCRIPT***

**Disusun dan diajukan oleh
INKA GUSTIANY MALLISA**

D421 14 517



**DEPARTEMEN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS HASANUDDIN
MAKASSAR**

2021

LEMBAR PENGESAHAN SKRIPSI

**IMPLEMENTASI INTEGRATED DEVELOPMENT ENVIRONMENT (IDE) BERBASIS
WEB UNTUK PEMROGRAMAN JAVASCRIPT**

Disusun dan diajukan oleh

INKA GUSTIANY MALLISA

D421 14 517

Telah dipertahankan di hadapan Panitia Ujian yang dibentuk dalam rangka Penyelesaian Studi

Program Sarjana Program Studi Teknik Informatika

Fakultas Teknik Universitas Hasanuddin

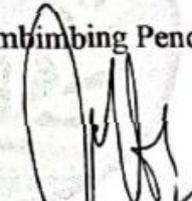
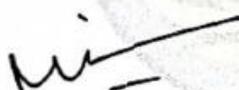
Pada tanggal 28 Juli 2021

dan dinyatakan telah memenuhi syarat kelulusan

Menyetujui,

Pembimbing Utama,

Pembimbing Pendamping,



Dr. Eng. Muhammad Niswar, S.T., M.IT.
NIP. 19730922 199903 1 001

Dr. Eng. Zukifli Tahir, S.T., M.Sc.
NIP. 19840403 201012 1 004



Ketua Program Studi

Dr. Amil Ahmad Alham, S.T., M.IT.
NIP. 19731010 199802 1 001

PERNYATAAN KEASLIAN

Yang bertanda tangan di bawah ini:

Nama : INKA GUSTIANY MALLISA
NIM : D421 14 517
Program Studi : TEKNIK INFORMATIKA
Jenjang : S1

Menyatakan dengan sebenar-benarnya bahwa skripsi yang berjudul:

IMPLEMENTASI *INTEGRATED DEVELOPMENT ENVIRONMENT (IDE)* BERBASIS WEB UNTUK PEMROGRAMAN *JAVASCRIPT*

Adalah karya tulis saya sendiri dan sepanjang pengetahuan saya di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah ditulis/diterbitkan sebelumnya, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan skripsi ini hasil karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Makassar, 18 Agustus 2021

Yang Membuat Pernyataan,



INKA GUSTIANY MALLISA

ABSTRAK

Perangkat lunak mulai berpindah dari aplikasi *desktop* ke berbasis *Web*. Layanan *online* dengan cepat mengganti produk perangkat lunak tradisional dengan produk yang langsung dapat diakses menggunakan internet. Dampaknya pun terjadi dalam lingkup pendidikan. *E-Learning* merupakan suatu pendekatan untuk mengembangkan pengetahuan, keterampilan dan kompetensi melalui sarana berbasis teknologi internet. Dengan adanya e-learning dapat membantu proses belajar mengajar agar lebih optimal. Salah satu penerapan *e-learning* ialah *Learning Management System (LMS)*. *Learning Management System (LMS)* merupakan sistem informasi yang membantu administrasi dan berfungsi sebagai *platform e-learning content*.

Dalam perkembangan *e-learning* saat ini, terlebih bagi pelajar jurusan teknik informatika, *LMS* yang disediakan oleh universitas masih tergolong sederhana. Oleh karena itu, dalam skripsi ini penulis membangun dan mengembangkan *LMS* baru dengan tujuan menyempurnakan *LMS* yang telah ada sebelumnya. Dimana pengembangan berfokus pada pembuatan fitur sistem *Integrated Development Environment (IDE)*. Selain itu, *LMS* yang dibangun memberikan fasilitas berupa pengumuman tenggang waktu tugas kuliah, bahan kuliah per kelas; seperti materi kuliah, kuis dan forum diskusi untuk pelajar dan penganjar (dosen).

Pengembangan *LMS* diimplementasikan menggunakan bahasa pemrograman *JavaScript* dengan *framework NodeJS*, begitu pula dengan pembuatan *IDE*. *IDE* ini dapat menerjemahkan bahasa pemrograman *HTML*, *CSS* dan *JavaScript*. Selanjutnya ketika sistem telah dibangun, akan dilakukan pengujian terkait analisa desain dan pengukuran kinerja sistem web. Dalam pengujian desain terdapat hasil kuisioner yang disebarkan kepada 25 pelajar mahasiswa semester awal sebagai responden. Kemudian disimpulkan bahwa *website LMS* yang dibangun sangat bermanfaat dan cukup mudah dimengerti pengaksesannya oleh *user* atau pelajar.

Kata Kunci: *web, Learning Management System (LMS), Integrated Development Environment (IDE), Javascript, NodeJS*

KATA PENGANTAR

Segala puji dan syukur kami panjatkan ke hadirat Tuhan Yang Maha Esa yang dengan anugerah dan kasih setia-Nya yang besar sehingga penulis dapat menyelesaikan tugas akhir dengan judul “*Implementasi Integrated Development environment (IDE) Berbasis Web Untuk Pemrograman JavaScript*” sebagai salah satu syarat dalam menyelesaikan jenjang Strata-1 pada Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin.

Dalam penyusunan skripsi ini, penulis menyadari bahwa skripsi ini masih jauh dari kata sempurna dan masih banyak kekurangan dikarenakan oleh segala keterbatasan dan kemampuan yang penulis miliki. Namun penulis berusaha untuk mempersembahkan skripsi ini sebaik-baiknya agar dapat memiliki manfaat bagi banyak pihak. Oleh karena itu, penulis menerima segala kritik dan saran yang membangun dalam perbaikan skripsi ini.

Penulis menyadari bahwa tanpa bantuan, dukungan dan bimbingan dari berbagai pihak, sampai dengan masa penyusunan tugas akhir ini, sangatlah sulit untuk menyelesaikannya. Oleh karena itu, pada kesempatan ini dengan ketulusan hati yang paling dalam, penulis mengucapkan terima kasih yang begitu besar kepada:

- 1) Tuhan Yang Maha Esa atas semua berkat, karunia serta pertolongan-Nya yang tiada batas, yang telah diberikan kepada penulis disetiap langkah dalam penulisan dan penyusunan skripsi ini.

- 2) Kedua orang tua tercinta, adik tersayang penulis, serta keluarga yang senantiasa memberikan kekuatan, inspirasi, motivasi, bimbingan moral, materi, kepercayaan dan kasih sayang yang tidak terbatas kepada penulis.
- 3) Bapak Dr.Eng. Muhammad Niswar, S.T., M.IT., selaku pembimbing 1 dan kepala Laboratorium *Ubiquitous Computing & Networking Lab* atau yang lebih dikenal dengan nama LAB UBICON yang telah banyak memberi bimbingan, inspirasi, motivasi, dan masukan yang bermanfaat selama masa pengerjaan tugas akhir penulis.
- 4) Bapak Dr.Eng Zulkifli Tahir, S.T.,M.Sc., selaku pembimbing II yang telah banyak memberi keyakinan, perhatian, bimbingan, motivasi, dan masukan yang bermanfaat kepada penulis.
- 5) Bapak Dr. Amil Ahmad Ilham, S.T., M.IT., selaku dosen penguji I sekaligus selaku Ketua Departemen Teknik Informatika dan Bapak A.Ais Prayogi Alimuddin, S.T., M.Eng., selaku dosen penguji yang telah memberikan saran sehingga laporan skripsi ini menjadi lebih baik.
- 6) Bapak Ir. Iqra Aswad, ST., MT., yang selalu menyediakan waktu, tenaga, pikiran dan perhatian yang luar biasa untuk mengarahkan penulis dalam penyusunan tugas akhir ini.
- 7) Bapak Robert dan Bapak Zainuddin serta segenap staf Departemen Teknik Informatika Fakultas Teknik Universitas Hasanuddin yang telah membantu kelancaran penyelesaian tugas akhir penulis.
- 8) Segenap keluarga LAB UBICON Universitas Hasanuddin yang telah memberikan begitu banyak bantuan selama penelitian, pengambilan data dan

diskusi *progress* penyusunan tugas akhir serta memberikan semangat di masa-masa sulit.

- 9) Para sahabat penulis yaitu FAMLONTS, sodara beda ibu kandung ade Nentha;Nelsa Indah Karambe, kaka Ikha; Friska S. Patty,S.E, ade Vii; Ayu Maqfiratul Jannah, ade Qhia; Andi Zasqia R. P. U. Basrah, Maxlonts; Max Aukila Sineri,S.E, dan La Faselan; Muhammad Faslan. Tiwi Nur Safitri,S.T, A. Khairil Fajri, Muh Zulfadli A. Suyuti, Gian Aron Angelo, Aryandi, S.T., Inka G. Mallisa, Fitriani Idrus, Winda Astiyanti, Armiani Putri, Rizka Irianti, David Reinhart, Zulfahmi, Fachrial yang telah selalu memberikan doa, bantuan dan dukungan kepada penulis.
- 10) Teman-teman dekat penulis di kampus Anastasia Yuki A. Sande, Tiwi Nur Safitri, Ulfah Rojiyah, Al Riefqy Dasmito, Maqhfira Putri Rahmat, Auraevadne, Cindy Oktaviani L.Bulan, Gian Aron Angelo, Muh. Zulfadli, Abdillah Satari Rahim, Hermawan Safrin, Yakip, Fitriani Idrus, Winda Astiyanti, Muh. Fachrial, Muh. Nur Alamsyah, Muh. Ardiansyah dan Syarif Hidayatullah.
- 11) Seluruh teman-teman RECTIFIER'14, KMKO ELEKTROTEKNIK'14 atas semua bantuan dan semangat yang diberikan selama ini.
- 12) Teman-teman ingame PUBG yang setiap main selalu mengingatkan akan pengerjaan skripsi dikarenakan masa usia perkuliahan saya.
- 13) Serta seluruh pihak yang tidak sempat disebutkan satu persatu yang telah banyak meluangkan tenaga, waktu, dan pikiran selama penyusunan laporan tugas akhir ini.

Akhir kata, penulis berharap semoga Tuhan Yang Maha Esa berkenan membalas segala kebaikan dari semua pihak yang telah banyak membantu. Semoga skripsi ini dapat memberikan manfaat bagi pengembangan ilmu selanjutnya. Amin.

Makassar, 7 Agustus 2020

Penulis

DAFTAR ISI

SKRIPSI	i
LEMBAR PENGESAHAN SKRIPSI	ii
PERNYATAAN KEASLIAN	iii
ABSTRAK	iv
KATA PENGANTAR	v
DAFTAR ISI	ix
DAFTAR GAMBAR	xii
DAFTAR TABEL	xiv
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Tujuan Penelitian	2
1.4. Manfaat Penelitian	3
1.5. Batasan Masalah	3
1.6. Sistematika Penulisan	4
BAB II TINJAUAN PUSTAKA	5
2.1 <i>Pengertian Website</i>	5
2.2 <i>Integrated Development Environment</i>	7
2.2.1 <i>Code Editor</i>	7

2.2.2	<i>Integrated Development Environment</i>	7
2.3	Bahasa Pemrograman	9
2.3.1	<i>JavaScript</i>	9
2.3.2	<i>NodeJS</i>	11
2.3.3	<i>PugJS</i>	12
2.4	Basis data.....	13
2.4.1	<i>MongoDB</i>	15
BAB III METODOLOGI PENELITIAN		18
3.1	Lokasi dan Waktu Penelitian.....	18
3.2	Instrumen Penelitian.....	18
3.3	Prosedur Penelitian.....	19
3.4	Tahap Persiapan.....	21
3.5	Gambaran Umum Sistem	22
3.5.1	<i>Use Case Diagram</i>	23
3.5.2	<i>Activity Diagram System</i>	29
3.6	Hasil Pembuatan Sistem dan pengujian <i>Black Box</i>	33
3.6.1	Hasil Pembuatan Sistem.....	33
3.6.1.1	Pengembangan Fitur <i>Integrated Development Environment</i> (<i>IDE</i>)	43
3.6.2	Pengujian <i>Black box</i>	50
3.7	Skenario Pengujian.....	53
3.7.1	Pengujian <i>Black box</i>	54
3.7.2	Pengujian Waktu Respon Website	54

3.7.3 Pengujian Penerapan UI (*User Interface*) dan UX (*User Experience*)

55

BAB IV HASIL DAN PEMBAHASAN.....	57
4.1 Pengujian <i>Black Box</i>	57
4.1.1 Hasil Pengujian	57
4.1.2 Pembahasan.....	60
4.2 Pengujian Waktu Respon Website	61
4.2.1 Hasil Pengujian	61
4.2.2 Pembahasan.....	62
4.3 Pengujian Penerapan UI (<i>User Interface</i>) dan UX (<i>User Experience</i>) ..	62
4.3.1 Hasil Pengujian	62
4.3.2 Pembahasan.....	64
BAB V PENUTUP	72
5.1 Kesimpulan.....	72
5.2 Saran	73
DAFTAR PUSTAKA	74
LAMPIRAN.....	76
LEMBAR PERBAIKAN SKRIPSI	116

DAFTAR GAMBAR

Gambar 2. 1. Konsep Sistem Basis data (Haidar Dzacko 2007).....	15
Gambar 2. 3 Cara Kerja MongoDB (Haidar Dzacko 2007).....	16
Gambar 2. 2 Terminologi dan konsep MongoDB dengan SQL (Haidar Dzacko 2007)	16
Gambar 3. 1 Diagram Tahapan Penelitian	19
Gambar 3. 2 Use case diagram system keseluruhan web.....	24
Gambar 3. 3 Use case diagram system pada modul Edit Exercise dalam pengaksesan Integrated Development Environment (IDE).....	27
Gambar 3. 4 Activity diagram menu fitur add exercise	30
Gambar 3. 5 Activity diagram menu fitur edit-update.....	31
Gambar 3. 6 Activity diagram menu fitur save-update pada code editor IDE.....	32
Gambar 3. 7 Activity diagram menu fitur see preview code dari code editor	33
Gambar 3. 8 Halaman beranda/home sistem - sebelum login (view dosen & mahasiswa).....	34
Gambar 3. 9 Halaman login (view dosen & mahasiswa).....	35
Gambar 3. 10 Halaman beranda/ <i>home</i> sistem - sesudah <i>login</i>	36
Gambar 3. 11 Halaman menu course list	37
Gambar 3. 12 Halaman macam-macam modul pada course.....	37
Gambar 3. 13 Halaman modul beserta isinya	38
Gambar 3. 14 Halaman exercise suatu course untuk menambahkan exercise baru	39

Gambar 3. 15 Halaman edit exercise untuk memberi intruksi perintah tugas yang akan diberikan	39
Gambar 3. 16 Halaman list mahasiswa yang telah mengerjakan exercise.....	40
Gambar 3. 17 Halaman preview hasil pengerjaan exercise dari mahasiswa pada IDE	41
Gambar 3. 18 Halaman beranda/home sistem – sesudah login (user sebagai mahasiswa).....	41
Gambar 3. 19 Halaman preview exercise beserta instruksi pengerjaan dari dosen	42
Gambar 3. 20 Halaman preview contoh exercise yang telah dikerjakan	43
Gambar 3. 21 Halaman tampilan tema warna yang dipilih pada code editor IDE di atas adalah ‘monokai’	45
Gambar 3. 22 Halaman tampilan tema warna yang dipilih pada code editor IDE di atas adalah ‘crimson editor’	46
Gambar 3. 23 Halaman <i>preview</i> penambahan fitur unduh data kode program (tombol-tombol fitur ini terletak tepat di samping tombol “Run”)	47
Gambar 3. 24 Hasil unduhan data dengan format <i>.html</i>	47
Gambar 3. 25 Hasil unduhan data dengan format <i>.css</i>	48
Gambar 3. 26 Hasil unduhan data dengan format <i>.js</i>	49
Gambar 4. 1 Grafik rata-rata hasil waktu akses	62
Gambar 4. 2 Rating scale dan jarak interval skala jawaban.....	65

DAFTAR TABEL

Tabel 3. 1 Pengujian black box fitur penambahan modul baru.....	50
Tabel 3. 2 Pengujian <i>black box</i> fitur pengeditan data	51
Tabel 3. 3 Pengujian <i>black box</i> fitur hapus data	51
Tabel 3. 4 Pengujian <i>black box</i> fitur lihat data.....	52
Tabel 3. 5 Pengujian <i>black box</i> fitur <i>publish/archive</i> data.....	52
Tabel 3. 6 Pengujian <i>black box</i> fitur <i>export</i> data	53
Tabel 3. 7 Pengujian <i>black box</i> fitur unggah gambar.....	53
Tabel 4. 1 Pengujian <i>black box</i> fitur penambahan modul baru.....	57
Tabel 4. 2 Pengujian <i>black box</i> fitur pengeditan data	58
Tabel 4. 3 Pengujian <i>black box</i> fitur hapus data	58
Tabel 4. 4 Pengujian <i>black box</i> fitur lihat data.....	59
Tabel 4. 5 Pengujian <i>black box</i> fitur <i>publish/archive</i> data.....	59
Tabel 4. 6 Pengujian <i>black box</i> fitur <i>export</i> data	60
Tabel 4. 7 Pengujian <i>black box</i> fitur unggah gambar.....	60
Tabel 4. 8 Tabel rata-rata hasil pengujian waktu respon pada web online	61
Tabel 4. 9 Hasil rekapitulasi jawaban responden.....	63
Tabel 4. 10 Penentuan nilai jawaban.....	64
Tabel 4. 11 Skor ideal	65
Tabel 4. 12 <i>Rating scale</i> dan jarak interval skala jawaban	65
Tabel 4. 13 Hasil <i>rating scale</i> pertanyaan 1	66
Tabel 4. 14 Hasil <i>rating scale</i> pertanyaan 2.....	66
Tabel 4. 15 Hasil <i>rating scale</i> pertanyaan 3.....	67

Tabel 4. 16 Hasil <i>rating scale</i> pertanyaan 4.....	67
Tabel 4. 17 Hasil <i>rating scale</i> pertanyaan 5.....	67
Tabel 4. 18 Hasil <i>rating scale</i> pertanyaan 6.....	68
Tabel 4. 19 Hasil <i>rating scale</i> pertanyaan 7.....	69
Tabel 4. 20 Hasil <i>rating scale</i> pertanyaan 8.....	69

BAB I

PENDAHULUAN

1.1. Latar Belakang

Pengembangan perangkat lunak adalah kegiatan penting di dunia saat ini. Seiring perkembangan komputer, ukuran dan kompleksitas produksi perangkat lunak meningkat. Dengan meningkatnya kompleksitas ini, menyelesaikan tugas-tugas seperti mengedit kode dan debugging mulai semakin sulit. Untuk itu, solusi dari masalah ini adalah dengan adanya *Integrated Development Environment (IDE)*. Perangkat lunak mulai berpindah dari aplikasi *desktop* ke berbasis *Web*. Layanan *online* dengan cepat mengganti produk perangkat lunak tradisional yang dapat di unduh dengan produk yang langsung dapat diakses menggunakan internet. *IDE* masih cukup tertinggal dalam hal perpindahan ke *web* namun terus berkembang.

Dalam pengenalan pembelajaran pemrograman, pelajar didorong berlatih menulis program untuk menguasai Bahasa dan sintaksisnya. Untuk itu, pelajar direkomendasikan untuk menginstal aplikasi *IDE (Integrated Development Environment)*. Beberapa *IDE*, terdapat persyaratan tambahan untuk menginstal *compiler/interpreter* dan *linker*. *IDE (Integrated Development Environment)* adalah sebuah program komputer yang memiliki beberapa fasilitas dalam pembuatan dan pengembangan perangkat lunak atau aplikasi. *IDE* merupakan tempat programmer menuliskan baris-baris kode serta menjalankan program [7]. Ada banyak *IDE* yang tersedia seperti seperti *NetBeans, Eclipse, Android Studio, Dev C++* dll.

Mayoritas *IDE* dimaksudkan untuk digunakan pada *desktop* ataupun laptop dan tentu saja tidak dimaksudkan untuk tablet dan perangkat portabel kecil lainnya. *IDE* pada umumnya merupakan program dengan kapasitas besar dan cukup berat yang disebabkan dari banyaknya fitur yang dimilikinya. Hal ini mengakibatkan sulitnya beberapa pelajar dalam hal penginstalan, yang dikarenakan terbatasnya spesifikasi personal komputer yang dimiliki[1]. *IDE* berbasis *web* dibuat agar para pelajar hanya fokus pada penulisan kode dan bukan pada penyuntingan/penginstalan framework lain. Dengan demikian *IDE* ini mampu memungkinkan pelajar dalam menyelesaikan setiap tugas pemrograman tanpa penyuntingan/penginstalan apapun[6].

Oleh karena itu, penulis mengangkat penelitian sesuai dengan judul “**Implementasi *Integrated Development Environment (IDE)* Berbasis Web untuk Pemrograman *JavaScript*”**. Dimana *IDE* ini dapat langsung diakses menggunakan *web browser* yang memiliki koneksi internet.

1.2. Rumusan Masalah

1. Bagaimana membangun sistem pembelajaran *online* pada *website*?
2. Bagaimana membuat dan mengintegrasikan fitur *IDE* pada *website*?
3. Bagaimana mengetahui efisiensi sistem pembelajaran *online* yang telah dibangun dalam penerapan antarmuka *website*?

1.3. Tujuan Penelitian

1. Mengetahui cara membangun dan mengembangkan sistem pembelajaran *online* pada *website*.
2. Mengetahui cara membuat dan mengintegrasikan fitur *IDE* pada *website*.

3. Mengetahui efisiensi dari sistem pembelajaran *online* yang telah dibangun dalam penerapan antarmuka *website*.

1.4. Manfaat Penelitian

1. Dapat mengetahui cara membangun dan mengembangkan sistem pembelajaran *online* pada *website*.
2. Dapat mengetahui cara membuat dan mengintegrasikan fitur *IDE* pada *website*.
3. Dapat mengetahui efisiensi dari sistem pembelajaran *online* yang telah dibangun dalam penerapan antarmuka *website*.

1.5. Batasan Masalah

Ruang lingkup pembahasan tugas akhir ini dibatasi hanya mencakup hal-hal berikut:

1. *Website* yang dibangun berupa *Learning Management System*.
2. Sistem *Website* yang dibangun menggunakan bahasa pemrograman *JavaScript* dan *PugJs*.
3. Fitur yang dimiliki dalam sistem web sama dengan *website e-learning* pada umumnya.
4. Sistem *IDE* yang dibuat menggunakan *NodeJs* dan *mochaJs*.
5. *IDE* yang dibuat hanya dapat menerjemahkan bahasa pemrograman *HTML*, *CSS* dan *JavaScript*.
6. Fitur *IDE* hanya mencakup pengeditan penulisan kode (*source code editor*), dan *compiler*.

7. Kinerja yang akan diukur adalah efisiensi dan performansi dari *website* yang akan dibangun.

1.6. Sistematika Penulisan

BAB I PENDAHULUAN: Bab ini berisi latar belakang masalah, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, sistematika penulisan.

BAB II TINJAUAN PUSTAKA: Pada bab ini akan dijelaskan teori-teori yang menunjang percobaan yang dilakukan.

BAB III METODOLOGI PENELITIAN: Bab ini berisi analisis kebutuhan sistem, perancangan sistem, dan skenario pengujian.

BAB IV HASIL DAN PEMBAHASAN: Bab ini berisi hasil penelitian dan pembahasan penjabaran dari penelitian yang dilakukan.

BAB V PENUTUP: Bab ini berisi kesimpulan dari hasil penelitian dan saran

BAB II

TINJAUAN PUSTAKA

2.1 Pengertian *Website*

Website atau situs merupakan kumpulan dari halaman-halaman yang berhubungan dengan file-file lain yang saling terkait. *Website* terdiri dari *page* atau halaman, salah satunya dikenal dengan sebutan *homepage*. *Homepage* adalah sebuah halaman yang pertama kali dilihat ketika seseorang mengunjungi sebuah *Website* dikarenakan *homepage* berada pada posisi teratas dengan halaman terkait yang berada di bawahnya. Halaman di bawah *homepage* disebut *child page* yang berisi *hyperlink* ke halaman lain dalam *web* (Gregorius, 2001).

Hyperlink adalah hubungan antar satu halaman *web* dengan halaman *web* lainnya. *Website* dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan *hyperlink*. Adapula teks yang dijadikan media penghubung disebut *hypertext* (Batubara 2015).

Website merupakan sekumpulan dokumen yang dipublikasikan melalui jaringan internet maupun intranet sehingga dapat diakses oleh *user* melalui *web browser* (Sardi, 2004).

Secara teknis, *web* adalah sebuah sistem dimana informasi dalam bentuk teks, gambar, suara, dan lain-lain yang tersimpan dalam sebuah internet *web server* dipresentasikan dalam bentuk *hypertext*. Informasi di *web* dalam bentuk

teks umumnya ditulis dalam format *HTML*. Informasi lainnya disajikan dalam bentuk grafis (dalam format GIF, JPG, PNG), suara (dalam format AU, WAV), dan objek multimedia lainnya (seperti MIDI, Shockwave, Quicktime Movie, 3D World). *Web* dapat diakses oleh perangkat lunak *web client* yang secara populer disebut sebagai *browser*. *Browser* membaca halaman-halaman yang tersimpan dalam *webserver* melalui protokol yang disebut HTTP (*Hypertext Transfer Protocol*). Sebagai dokumen *hypertext*, dokumen-dokumen di *web* dapat memiliki *link* dengan dokumen lain, baik yang tersimpan dalam *web server* yang sama maupun di *web server* lainnya. Link memudahkan para pengakses *web* berpindah dari satu halaman ke halaman lainnya, dan "berkelana" dari satu server ke server lain. Kegiatan penelusuran halaman *web* ini biasa diistilahkan sebagai *browsing*, ada juga yang menyebutnya sebagai *surfing* (berselancar). Seiring dengan semakin berkembangnya jaringan internet di seluruh dunia, maka jumlah situs *web* yang tersedia juga semakin meningkat. Hingga saat ini, jumlah halaman *web* yang bisa diakses melalui internet telah mencapai angka miliaran (Batubara 2015).

Pengembangan *website* adalah salah satu aspek yang paling penting dari situs *web* apapun, sebagai pengembangan dari situs yang bekerja pada *coding* dan pemrograman untuk membuat *web page* bekerja dengan baik. Sementara pembangun *site* gratis dapat hadir dengan cara membuat *site* sendiri. Tidak hanya pada *website* yang akan dikembangkan sesuai selera, tetapi akan dibuat sehingga dapat dengan mudah bekerja dengan *CMS* untuk menjalankan semua aspek yang berbeda dari *website* yang telah dibuat. *Website* dinamis adalah sebuah *website*

yang isi kontennya dapat berubah mengikuti dari isi *database*. Seseorang tidak perlu mengganti pemrograman *website* melainkan cukup melakukan pembaharuan pada *database* yang digunakan (Rahman, 2013).

Untuk memudahkan penelusuran halaman *web*, terutama untuk menemukan halaman yang memuat topik-topik yang spesifik, maka para pengakses *web* dapat menggunakan suatu mesin pencari (*search engine*). Penelusuran berdasarkan *search engine* dilakukan berdasarkan kata kunci (*keyword*) yang kemudian akan dicocokkan oleh *search engine* dengan basis data miliknya (Batubara 2015).

2.2 *Integrated Development Environment*

IDE adalah lingkup pemrograman yang diintegrasikan kedalam suatu aplikasi perangkat lunak yang menyediakan pembangun *Graphic User Interface (GUI)*, suatu text atau kode editor, suatu compiler atau interpreter dan suatu debugger.

2.2.1 *Code Editor*

Code editor adalah tempat dimana kita meletakkan atau menuliskan kode program dari program aplikasi kita. Pada *code editor* terdapat bagian objek dan *event* dari *control*. Pada bagian objek terdapat semua objek yang digunakan pada proyek.

2.2.2 *Integrated Development Environment*

Code editor mempunyai kelemahan yaitu tidak mempunyai kemampuan untuk menjalankan program yang telah dibuat. Oleh sebab itu, muncul teknologi *IDE (Integrated Development Environment)*. Dengan

menggunakan *IDE*, *programmer* dapat langsung menjalankan kode program tanpa perlu mengkompilasi dan mengeksekusi lewat *command-line*.

Integrated Development Environment adalah aplikasi yang menyediakan fasilitas bagi *programmer* untuk pengembangan perangkat lunak seperti melengkapi dan memperbaiki kode, mengedit dan mengelola *source code*, pengujian otomatis, dll. *IDE* berbasis *desktop* yang modern mengintegrasikan berbagai alat rekayasa perangkat lunak, menyediakan *platform* untuk menulis, memelihara, menguji, membangun, menjalankan dan debugging. Mereka meningkatkan produktifitas pengembang dengan memasukkan berbagai jenis layanan editor khusus untuk syntax dan semantic Bahasa. Layanan ini membantu pengembang dalam memahami dan menavigasikan kode, serta mengarahkan pengembang ke area kode yang tidak konsisten atau tidak lengkap dan mereka bahkan membantu dengan mengedit kode dengan memberikan lekukan otomatis, penyisipan bracket dan penyelesaian konten. Sistem lokal *IDE* diinstal pada suatu sistem. *IDE* berdiri sendiri dengan sumber daya computer yang tinggi karena *IDE* mendukung lebih banyak fasilitas serta fitur kepada *programmer*. Oleh sebab itu, *IDE* membutuhkan lebih banyak sumber daya komputer, terutama memori dan *CPU*, yang mungkin tidak tersedia setiap saat. Sebagian besar *IDE* berbasis *desktop* membutuhkan lingkungan pengembangan yang harus diatur pada mesin mereka. Lingkungan pengembangan ini membutuhkan *IDE* Bahasa spesifik seperti *eclipse* atau *Sublime Text* untuk diunduh dan dikonfigurasi dalam mesin pengguna.

2.3 Bahasa Pemrograman

Bahasa pemrograman, atau sering diistilahkan juga dengan Bahasa komputer, adalah instruksi standar untuk memerintah komputer. Bahasa pemrograman ini merupakan suatu himpunan dari aturan sintaks dan semantik yang dipakai untuk mendefinisikan program komputer. Adapun Bahasa pemrograman yang digunakan dalam pengerjaan sistem ini adalah *JavaScript* dengan menggunakan *Framework NodeJS*, dan *PugJs*.

2.3.1 *JavaScript*

JavaScript adalah bahasa pemrograman yang populer. *JavaScript* adalah bahasa pemrograman yang digunakan untuk *HTML* dan *web*, untuk Server, PC, laptop, tablet dan lebih banyak lagi. Kode pemrograman *JavaScript* dapat disisipkan kedalam halaman *HTML*. Pada awalnya, *JavaScript* mulai diperkenalkan di *browser* Netscape Navigator 2. Namun waktu itu namanya bukan *JavaScript*, namun *LiveScript*. Mengingat pada waktu itu teknologi Java sedang panas-panasnya atau sedang tren, maka pihak Netscape memutuskan untuk mengganti namanya menjadi *JavaScript*, yang sepertinya nama tersebut lebih *marketible* dibandingkan *LiveScript*. Selanjutnya pihak Microsoft (rival Netscape) pun mulai ikut-ikutan memfasilitasi *web browser* buatannya, '*Internet Explorer*', supaya bisa mendukung *JavaScript*. Namun mungkin karena gengsi, pihak Microsoft memberi nama bahasa yang lain, yaitu *JavaScript*. Mulai saat itu, Netscape dan Microsoft mulai berlomba-lomba mengembangkan bahasa tersebut dalam versi yang berlainan. Oleh sebab persaingan itulah terkadang suatu *JavaScript*

mungkin bisa bekerja dengan baik di *browser* Netscape, tapi tidak demikian halnya di IE, begitu pula sebaliknya (Permana 2016).

Pada awalnya, *JavaScript* mulai diperkenalkan di *browser* Netscape Navigator 2. Namun waktu itu namanya bukan *JavaScript*, namun *LiveScript*. Mengingat pada waktu itu teknologi Java sedang panas-panasnya atau sedang tren, maka pihak Netscape memutuskan untuk mengganti namanya menjadi *JavaScript*, yang sepertinya nama tersebut lebih marketible dibandingkan *LiveScript*. Selanjutnya pihak Microsoft (rival Netscape) pun mulai ikut-ikutan memfasilitasi *web browser* buatannya, 'Internet Explorer', supaya bisa mendukung *JavaScript*. Namun mungkin karena gengsi, pihak Microsoft memberi nama bahasa yang lain, yaitu Jscript. Mulai saat itu, Netscape dan Microsoft mulai berlomba-lomba mengembangkan bahasa tersebut dalam versi yang berlainan. Oleh sebab persaingan itulah terkadang suatu *JavaScript* mungkin bisa bekerja dengan baik di *browser* Netscape, tapi tidak demikian halnya di IE, begitu pula sebaliknya (Permana 2016).

Ada dua jenis bagaimana *JavaScript* dibuat, pertama *JavaScript* ditulis dalam file yang terpisah dengan *HTML*, kedua *JavaScript* ditulis dalam *HTML*. *JavaScript* yang ditulis diluar *HTML* disebut Eksternal *JavaScript* dengan ekstensi file .js. Dalam *HTML*, penulisan script diawali dengan. Script yang akan dijalankan harus diletakkan diantara <script> dan </script> tag <script> memiliki beberapa atribut, namun yang terpenting adalah atribut language dan type. Karena *JavaScript* bukan satu – satunya bahasa scripting, maka sangatlah perlu untuk memberitahukan kepada *browser* bahwa bahasa

script yang digunakan adalah *JavaScript* dan selanjutnya *browser* akan menjalankan modul pendukung *JavaScript* untuk memprosesnya (Permana 2016).

2.3.2 NodeJS

NodeJS merupakan platform server yang dibangun menggunakan *JavaScript* dan berjalan di dalam interpreter Chrome *JavaScript* runtime. Dibuat untuk pengembangan perangkat lunak berbasis *web* dengan cepat, aplikasi jaringan yang scalable. *NodeJS* menggunakan event-driven, model non-blocking I/O yang membuatnya menjadi ringan dan efisien. Sangat baik digunakan untuk aplikasi waktu-nyata yang digunakan diberbagai perangkat (Muhammad Agung Rizkyana 2014) .

NodeJS adalah perangkat lunak yang didesain untuk mengembangkan aplikasi berbasis *web* dan ditulis dalam sintaks bahasa pemrograman *JavaScript*. Bila selama ini *JavaScript* dikenal sebagai bahasa pemrograman yang berjalan di sisi *client / browser* saja, maka *NodeJS* ada untuk melengkapi peran *JavaScript* sehingga bisa juga berlaku sebagai bahasa pemrograman yang berjalan di sisi server, seperti halnya *PHP*, *Ruby*, *Perl* dan sebagainya. *NodeJS* dapat berjalan di sistem operasi *Windows*, *Mac OS X* dan *Linux* tanpa perlu ada perubahan kode program. *NodeJS* memiliki pustaka server *HTTP* sendiri sehingga memungkinkan untuk menjalankan server *web* tanpa menggunakan program server *web* seperti *Apache* atau *Nginx*.

NodeJS dibuat dengan *engine* yang sama dengan *browser chrome* yang dikembangkan oleh google yang bersifat *open source* dimana *NodeJS*

memiliki keuntungan pada sifatnya yang *non blocking*. Sebagai contoh misalnya pada bahasa program biasa bila terdapat sebuah fungsi A yang berjalan, maka umumnya fungsi A harus selesai terlebih dahulu kemudian menjalankan fungsi B. tetapi berbeda dengan *NodeJS* yang sifatnya parallel, artinya *NodeJS* akan mengerjakan hal – hal yang sama dan ketika sudah selesai dengan fungsi yang ada maka dapat ditambahkan sebuah fungsi *callback*. *Callback* ini yang akan melihat apakah tugas A sudah selesai dikerjakan dan tidak akan menunggu A selesai tetapi juga akan sambil menjalankan tugas B.

Jadi pada program yang lain sebuah proses akan diselesaikan terlebih dahulu baru dapat ke proses berikutnya. Tapi pada *NodeJS* berbeda karena dia berbasis *JavaScript* maka dia tidak akan menunggu proses A itu selesai tetapi semua proses akan dijalankan satu-satu lalu akan menjalankan proses berikutnya yang kita sudah tentukan. Dengan kata lain *NodeJS* bisa mengatasi *multiple request* secara bersamaan dan prosesnya itu tidak langsung diselesaikan tetapi menjalankan beberapa proses terlebih dahulu.

2.3.3 PugJS

Beberapa *template engine* yang populer digunakan di *expressJs* adalah *PugJS*, *Mustache* dan *EJS*. *Template engine* bekerja dengan menginjeksi data dan menghasilkan *HTML* untuk dibaca oleh *browser*. *PugJS* adalah *template engine* yang diimplementasikan dengan *JavaScript* untuk *NodeJS* dan *web browser*. Fungsi utama dari *PugJS* adalah untuk memproses *template* dan konten menjadi dokumen *HTML*. Berkas *template pugJS* (disimpan dengan

ekstensi *.pug*) ditulis dengan menggunakan *syntax* yang lebih sederhana dari *HTML*.

PugJS merupakan sebuah *library* untuk templating *HTML*. Melalui *PugJS*, *user* dapat membagi halaman *HTML* menjadi komponen-komponen kecil yang bersifat *reusable* atau bisa digunakan di berbagai tempat, di samping itu *PugJS* juga memungkinkan untuk menjalankan *conditional* dan *looping*.

Proses rendering *PugJS* umumnya sederhana, *pug.compile()* akan mengkompilasi source code *PugJS* menjadi fungsi *JavaScript* yang mengambil objek data (disebut "*locals*") sebagai argument. Memanggil fungsi yang dihasilkan dengan data *user* dan mengembalikan *string HTML* yang diberikan dari data *user*. Fungsi yang dikompilasi dapat digunakan kembali, dan dipanggil dengan set data berbeda. *PugJS* juga menyediakan *pug.render()* yang berfungsi untuk menggabungkan kompilasi dan *rendering* menjadi satu langkah. Namun, fungsi *template* akan dikompilasi ulang setiap kali *render* dipanggil, yang dapat memengaruhi kinerja. Kalau tidak, *user* dapat menggunakan opsi *cache* dengan *render*, yang secara otomatis akan menyimpan fungsi yang dikompilasi ke dalam *cache internal*.

2.4 Basis data

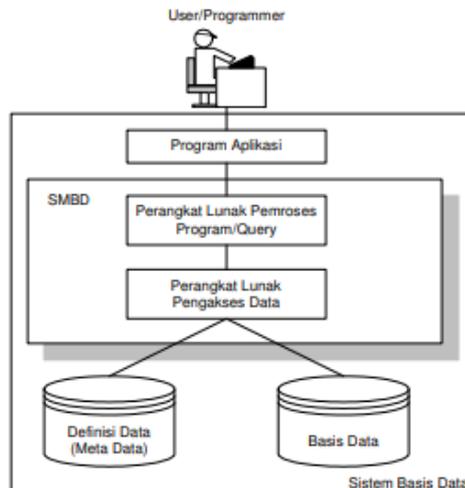
Data merupakan fakta mengenai suatu objek seperti manusia, benda, peristiwa, konsep, keadaan dan sebagainya yang dapat dicatat dan mempunyai arti secara implisit. Data dapat dinyatakan dalam bentuk angka, karakter atau simbol, sehingga bila data dikumpulkan dan saling berhubungan maka dikenal dengan

istilah basis data (basis data). Sedangkan menurut George Tsu-der Chou basis data merupakan kumpulan informasi bermanfaat yang diorganisasikan ke dalam aturan yang khusus. Informasi ini adalah data yang telah diorganisasikan ke dalam bentuk yang sesuai dengan kebutuhan seseorang. Menurut *Encyclopedia of Computer Science and Engineer*, para ilmuwan di bidang informasi menerima definisi standar informasi yaitu data yang digunakan dalam pengambilan keputusan. Definisi lain dari basis data menurut Fabbri dan Schwab adalah sistem berkas terpadu yang dirancang terutama untuk meminimalkan duplikasi data. Menurut Ramez Elmasri mendefinisikan basis data lebih dibatasi pada arti implisit yang khusus, yaitu (Haidar Dzacko 2007):

- a. Basis data merupakan penyajian suatu aspek dari dunia nyata (*real world*).
- b. Basis data merupakan kumpulan data dari berbagai sumber yang secara logika mempunyai arti implisit. Sehingga data yang terkumpul secara acak dan tanpa mempunyai arti, tidak dapat disebut basis data.
- c. Basis data perlu dirancang, dibangun dan data dikumpulkan untuk suatu tujuan. Basis data dapat digunakan oleh beberapa *user* dan beberapa aplikasi yang sesuai dengan kepentingan *user*.

Dari beberapa definisi-definisi tersebut, dapat dikatakan bahwa basis data mempunyai berbagai sumber data dalam pengumpulan data, bervariasi derajat interaksi kejadian dari dunia nyata, dirancang dan dibangun agar dapat digunakan oleh beberapa pengguna untuk berbagai kepentingan.

C. J. Date menyatakan bahwa sistem basis data dapat dianggap sebagai tempat untuk sekumpulan berkas data yang terkomputerisasi dengan tujuan untuk memelihara informasi dan membuat informasi tersebut tersedia saat dibutuhkan (Haidar Dzacko 2007).



Gambar 2. 1. Konsep Sistem Basis data (Haidar Dzacko 2007)

Basis data yang membutuhkan sebuah media untuk melakukan penyimpanan dan pengelolaan data yang ada yang diolah oleh program *SQL* maupun *NoSQL*. Keduanya adalah media yang digunakan untuk melakukan pengolahan basis data yang digunakan sesuai dengan kebutuhan pengembang.

2.4.1 *MongoDB*

MongoDB adalah salah satu jenis *database NoSQL* yang berbasis dokumen dengan format *JSON*. Seperti arti dari namanya, *database* ini merupakan *database* yang tidak memakai bahasa *SQL query* data tapi bisa secara langsung menggunakan bahasa pemrograman *client*. Hal ini tentu berbeda dengan *database* yang menggunakan konsep *MySQL* dengan *relational database management system (RDBMS)*-nya. Dimana *database* ini mengutamakan entitas dari tabel yang

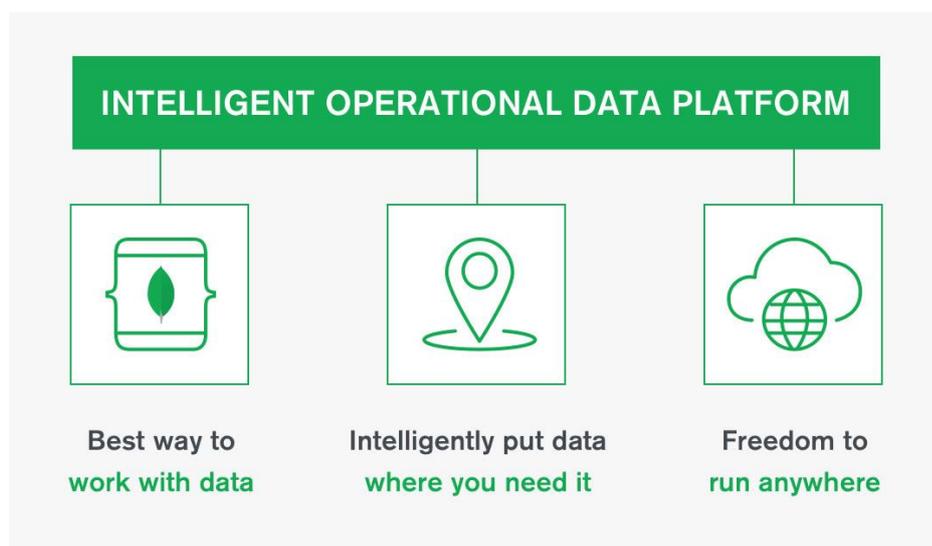
memiliki keterkaitan satu sama lain atau relasional *database*. Sedangkan *database*

SQL Terminologi / Konsep	Istilah MongoDB / konsep	Penjelasan / Keterangan
Database	Database	database
tabel	koleksi	tabel database / Koleksi
baris	dokumen	Baris perekaman data / Dokumentasi
kolom	bidang	Data Field / Domain
indeks	indeks	indeks
tabel bergabung		Tabel bergabung, MongoDB tidak mendukung
kunci utama	kunci utama	kunci utama, MongoDB otomatis <code>_id</code> lapangan sebagai kunci utama

Gambar 2. 3 Terminologi dan konsep *MongoDB* dengan SQL (Haidar Dzacko 2007)

NoSQL tidak memiliki relasi atau non-relasional antar kabel serta data yang disimpan tidak dalam format tabel. Berikut adalah terminologi dan konsep di *MongoDB* dan perbandingan dengan konsep yang ada di *SQL database*.

Teknologi *NoSQL* memiliki keunggulan sendiri dalam melakukan manajemen basis data sehingga penggunaan jenis basis data ini lebih diminati oleh pengembang. *Database NoSQL* seperti *MongoDB* menyimpan data



Gambar 2. 2 Cara Kerja *MongoDB* (Haidar Dzacko 2007)

sebagai dokumen yang *schema-less* yang artinya data yang disimpan

mempunyai *key - value* yang tidak terikat atau bebas. Klien bisa berinteraksi langsung dengan *database MongoDB* dengan menggunakan *shell JavaScript “mongo “* untuk administrasi dan *query* data (Haidar Dzacko 2007).