

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1. Tinjauan Pustaka

Dalam membuat aplikasi ini merujuk dari lima penelitian yang sudah ada sebelumnya. Pertama, penelitian Akhmad Alimudin, Edi Satriyanto, S.Si, M.Si dan Idris Winarno, S.ST,M.Kom tahun 2010 dengan judul “Perancangan dan Pembuatan CMS *Tryout Online* pada Dinas Pendidikan dengan menggunakan *Database Terdistribusi*”. Kedua, penelitian Albert Leonardo Pisa dkk tahun 2015 yang berjudul “Aplikasi *Audience Response System* Berbasis Web dan Android”. Ketiga, penelitian Muhammad Agung Rizkiyana, R Sandhika Galih Amalga tahun 2014 yang berjudul “Rancangan Arsitektur Aplikasi Pengumpulan Tugas Dengan *Push Notification Real-Time* menggunakan Node.js”. Keempat penelitian Muhammad Iqbal C. R dkk tahun 2012 yang berjudul “Implementasi Klien SIP Berbasis Web menggunakan Html5 dan Node.js”. Kelima Tugas Akhir Nafsiatul Afiyah tahun 2014 yang berjudul “Perancangan Aplikasi *Tryout* Bimbingan Belajar Berbasis Web”.

Perbandingan penelitian sebelumnya dengan penelitian saya dapat dilihat di

Tabel 2.1 berikut:

Tabel 2.1. Tinjauan Pustaka

Parameter Penulis	Judul	Objek	Teknologi	Output
Akhmad Alimudin, Edi Satriyanto, S.Si, M.Si, Idris Winarno, S.ST,M.Kom (2010)	Perancangan dan Pembuatan CMS <i>Tryout Online</i> pada Dinas Pendidikan dengan menggunakan <i>Database Terdistribusi</i>	<i>Tryout</i>	Web Cms	Web Cms <i>Tryout Online</i>
Albert Leonardo Pisa, dkk (2015)	Aplikasi <i>Audience Response System</i> Berbasis Web dan Android	<i>Audience Response</i>	Node.js	<i>Poling realtime</i>
Muhammad Agung Rizkiyana, R Sandhika Galih Amalga (2014)	Rancangan Arsitektur Aplikasi Pengumpulan Tugas Dengan <i>Push Notification Real-Time</i> menggunakan Node.js	Pengumpulan Tugas	Node.js	notifikasi telah <i>upload</i> tugas, notifikasi tugas dinilai, estimasi waktu.
Muhammad Iqbal C. R dkk (2012)	Implementasi Klien SIP Berbasis Web menggunakan Html5 dan Node.js	Klien SIP	Node.js	Komunikasi <i>Audio</i> dan video

Tabel 2.2. Lanjutan Tinjauan Pustaka

Parameter Penulis	Judul	Objek	Teknologi	Output
Nafsiatul Afiyah (2014)	Perancangan Aplikasi <i>Tryout</i> Bimbingan Belajar Berbasis Web	<i>Tryout</i>	-	Nilai Hasil Ujian <i>Tryout</i> siswa, artikel.
Alwigani Rachman Badawi	Aplikasi Simulasi <i>Tryout</i> Ujian Nasional Menggunakan Node.Js	<i>Tryout</i>	<i>Tryout</i>	Nilai Hasil <i>Tryout</i> siswa, Nilai rata-rata

2.2 Dasar Teori

2.2.1 Javascript

Javascript merupakan bahasa pemrograman yang rilis pada tahun 1995. Meskipun memiliki nama *javascript*, bahasa ini berbeda dengan bahasa pemrograman java. Ada beberapa catatan mengenai *javascript*, antara lain:

1. *Javascript* adalah bahasa scripting. *Javascript* adalah program script yang dibaca dan dieksekusi menggunakan *interpreter*. Berbeda dengan bahasa pemrograman yang dikompilasi, dimana setiap program dibaca oleh *compiler* dan di translasikan kedalam bentuk *executable file*.
2. Seperti bahasa C. Dasar dari *syntax* dan struktur *javascript* mengacu pada bahasa pemrograman C.

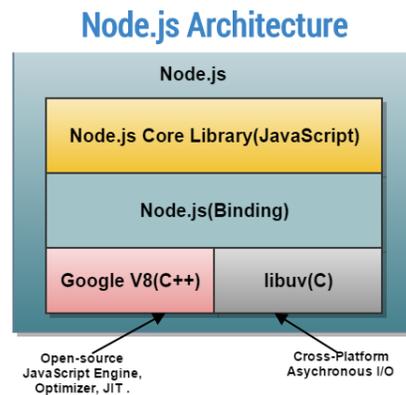
3. Bahasa berorientasi objek. Inti dari *javascript* mengandung kumpulan inti objek–objek, seperti *array*, *date*, dan *math* serta kumpulan inti dari bahasa pemrograman seperti operator, struktur kendali dan *statements*. Inti dari *javascript* dapat diturunkan untuk memenuhi beberapa tujuan dengan menambahkan beberapa objek–objek tambahan pada inti *javascript* (Muhammad Agung Rizkyana dkk,2014).

2.2.2 Node.js

Node.js merupakan sebuah platform untuk membuat aplikasi Javascript yang dapat dijalankan di sisi server. Node.js dikembangkan dari engine Javascript yang dibuat oleh Google untuk browser Chrome bernama V8. Node.js menggunakan Javascript sebagai bahasa pemrograman dan teknik event-driven, nonblocking I/O (asynchronous) yang membuatnya lebih ringan dan efisien. Node.js memiliki fitur built-in HTTP server library yang mampu menjadikannya sebuah web server tanpa bantuan software lainya seperti Apache dan Nginx.

Pada dasarnya, Node.js adalah sebuah *runtime environment* dan *script library*. Sebuah *runtime environment* adalah sebuah software yang berfungsi untuk mengeksekusi, menjalankan dan mengimplementasikan fungsi-fungsi serta cara kerja inti dari suatu bahasa pemrograman. Sedangkan *script library* adalah kumpulan kompilasi atau bank data yang berisi skrip atau kode-kode pemrograman (Equan Pr, 2013).

Node.js dibangun dengan menggunakan *JavaScript* dan *C/C++*, adapun arsitektur node.js seperti pada **Gambar 2.1**. Google V8 dalam arsitektur node.js berfungsi sebagai mesin *compiler* yang ditulis dalam *C++* dan *Library Libuv* bertanggung jawab untuk menangani operasi *asynchronous I/O* dan *loop* utama.



Gambar 2.1 Arsitektur node.js

Sumber: <http://www.assignmenthelp.net/node-js>

2.2.3 MongoDB (*NoSQL*)

MongoDB adalah sebuah *document oriented database* yang bersifat *open source*. MongoDB merupakan salah satu *database NoSQL* yang memiliki sebuah konsep penyimpanan data *non-relational*. Istilah *NoSQL* merupakan kepanjangan dari “*Not Only SQL*” yaitu sistem manajemen database yang berbeda dari sistem manajemen database rasional dalam beberapa cara. Penyimpanan data tanpa perlu adanya tabel schema dan tidak ada bahasa sql yang terlibat dalam pemakaian database.

Dalam MongoDB tidak mengenal adanya tabel, kolom dan baris jadi tidak ada schema dalam MongoDB (*schema-less*). Unit paling kecil pada dari MongoDB adalah *document*, sedangkan kumpulan dari *document* adalah *collection*. Seperti halnya dalam database rasional *document*, ibarat sebuah record dan *collection* pada sebuah tabel *document* dalam MongoDB dapat memiliki atribut yang berbeda-beda dengan *document* yang lainnya walaupun pada satu *collection*.

MongoDB dibuat dengan menggunakan C++, yang memiliki beberapa fitur yaitu *Document-oriented storage, full index, support replication dan high availability, auto-sharding, querying, fast in-place updates, map reduce, gridFS, commercial support*. Pada MongoDB tidak membutuhkan struktur tabel jadi untuk melakukan proses *insert* maka MongoDB akan otomatis membuatkan struktur tabelnya. MongoDB mempunyai performa 4x lebih cepat jika dibandingkan dengan Mysql. MongoDB dikembangkan sejak Oktober 2007 oleh 10Gen dan dirilis ke *public* sejak Februari 2009 yang mempunyai lisensi GNU AGPL 3.0 dan *Apache License* untuk drivernya.

2.2.4 Framework ExpressJS

Framework adalah kerangka kerja yang terdiri dari kumpulan kelas dan fungsi yang disusun secara sistematis, sehingga dapat digunakan untuk membantu membuat aplikasi utuh tanpa harus membuat semua kodenya dari awal. Secara umum *framework* menggunakan struktur *MVC*. Dalam pembuatan aplikasi menggunakan peranti lunak Node.js ada berbagai *framework* yang dapat digunakan salah satunya adalah *express.js*.

Express.js adalah kerangka aplikasi web Node.js yang minimal dan fleksibel yang menyediakan seperangkat fitur untuk mengembangkan web dan aplikasi mobile.

Berikut ini adalah beberapa fitur inti dari kerangka Express.js:

- a) Memungkinkan untuk mengatur middlewares untuk menanggapi Permintaan HTTP.
- b) Mendefinisikan sebuah tabel routing yang digunakan untuk melakukan tindakan yang berbeda berdasarkan metode HTTP dan URL.
- c) Memungkinkan untuk secara dinamis membuat Pages HTML berdasarkan argumen untuk template.

2.2.5 NPM (*Network Printer Manager*)

NPM merupakan paket *manager* untuk Node.js, yang ditemukan pada tahun 2009 sebagai suatu proyek terbuka untuk membantu pengembang *Javascript* saling berbagi kode paket modul. NPM juga merupakan sebuah kode perintah untuk memungkinkan pengembang menggunakan dan mempublikasi paket modul.

2.2.6 *Tryout*

Tryout menurut Suparman Usup, (2012) dalam penelitiannya yang berjudul “Perancangan Sistem Informasi *Tryout Online* Pada SMA Nasional Bandung” menyimpulkan bahwa *tryout* adalah suatu mekanisme yang digunakan sebagai sebuah latihan bagi siswa sebelum melaksanakan ujian yang sesungguhnya. Sedangkan *Tryout* Pra Ujian Nasional adalah sebuah uji coba (*tryout*) yang diadakan untuk mengetahui kemampuan siswa dalam menjawab soal Ujian Nasional periode sebelumnya.

2.2.7 *UML*

UML (Unified Modelling Language) adalah suatu bahasa yang sudah menjadi *standart* pada *visualisasi*, perancangan, dan pendokumentasian sistem *software* yang merupakan suatu metode pemodelan secara visual untuk sarana perancangan sistem berorientasi objek. Beberapa tujuan atau fungsi dari penggunaan *UML*, antara lain:

1. Dapat memberikan bahasa permodelan visual kepada pengguna dari berbagai macam pemrograman maupun proses rekayasa.
2. Dapat menyatukan praktek-praktek terbaik yang ada dalam permodelan.
3. Dapat memberikan model yang siap untuk digunakan, merupakan bahasa permodelan visual yang ekspresif untuk mengembangkan sistem dan untuk saling menukar model secara mudah.
4. Dapat berguna sebagai blue print, sebab sangat lengkap dan detail dalam perancangannya yang nantinya akan diketahui informasi yang detail mengenai koding suatu program.
5. Dapat memodelkan sistem yang berkonsep berorientasi objek, jadi tidak hanya digunakan untuk memodelkan perangkat lunak (*software*) saja.
6. Dapat menciptakan suatu bahasa permodelan yang nantinya dapat dipergunakan oleh manusia maupun oleh mesin (Sora N, 2015).

Aplikasi simulasi *tryout* ini menggunakan empat diagram *UML* diantaranya yaitu:

1. Use case diagram

Use case diagram yaitu salah satu jenis diagram pada *UML* yang menggambarkan interaksi antara sistem dan aktor, *use case diagram* juga dapat men-deskripsikan tipe interaksi antara si pemakai sistem dengan sistemnya.

2. Activity Diagram

Activity diagram atau diagram aktivitas yaitu salah satu jenis diagram pada *UML* yang dapat memodelkan proses-proses apa saja yang terjadi pada sistem.

3. Sequence diagram

Sequence diagram yaitu salah satu jenis diagram pada *UML* yang menjelaskan interaksi objek yang berdasarkan urutan waktu, *sequence diagram* juga dapat menggambarkan urutan atau tahapan yang harus dilakukan untuk dapat menghasilkan sesuatu seperti pada *use case diagram*.

4. Class diagram

Class diagram yaitu salah satu jenis diagram pada *UML* yang digunakan untuk menampilkan kelas-kelas maupun pakaet-paket yang ada pada suatu sistem yang nantinya akan digunakan. Jadi diagram ini dapat memberikan sebuah gambaran mengenai sistem maupun relasi-relasi yang terdapat pada sistem tersebut.