

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Dalam tinjauan pustaka di bawah ini terdapat beberapa referensi yang dapat dijadikan acuan untuk penelitian, referensi tersebut adalah sebagai berikut:

David Karismata W. Tion (2016) membuat bot Telegram untuk Remote Monitoring pada server. Aplikasi ini berfungsi untuk mengawasi kinerja dari server serta memberikan notifikasi kepada Server Administrator jika terjadi masalah.

Gde Sastrawangsa (2017) melakukan penelitian mengenai pemanfaatan Telegram Bot yang diintegrasikan dengan Sistem Informasi Akademik. Sistem ini berfungsi untuk memberikan layanan dan informasi akademik melalui Telegram Bot.

Tomi Bong (2017) melakukan penelitian mengenai pemanfaatan Telegram Bot untuk mengakses portal akademik. Sistem ini berfungsi untuk memberikan informasi akademik melalui Telegram Bot.

Raga Nufusula dan Ajib Susanto (2018) membuat Chat Bot untuk server pulsa menggunakan Telegram API. Aplikasi ini berguna untuk menggantikan SMS dengan Bot Telegram untuk isi ulang pulsa prabayar.

Yoga Dwi Jatmika (2018) membuat aplikasi E-Commerce dengan menggunakan *framework* Codeigniter serta MySQL sebagai teknologi basisdata. Aplikasi ini mengolah pemesanan bahan bangunan pada U.D. Karya Utama.

Perbandingan antara beberapa penelitian yang digunakan sebagai tinjauan pustaka bisa dilihat pada Tabel 2.1.

Tabel 2. 1 Perbandingan Penelitian

Peneliti	Judul	Metode / Teknologi	Bahasa Pemrograman	Studi Kasus
David Karismata W. Tion (2016)	Perancangan Bot untuk Remote Monitoring pada Server menggunakan Telegram Bot API	Telegram Bot API, PPDIIO	Python	Remote Monitoring Server
Gde Sastrawangsa (2017)	Pemanfaatan Telegram Bot Untuk Automatisasi Layanan Dan Informasi Mahasiswa Dalam Konsep Smart Campus	Telegram Bot API, Webhooks	PHP, SQL	Sistem Informasi Akademik
Tomi Bong (2017)	Pembangunan Aplikasi Media Informasi Mahasiswa Berbasis Bot Telegram	Telegram Bot API, Waterfall	PHP	Media Informasi Mahasiswa
Raga Nufusula, Ajib Susanto (2018)	Rancang Bangun Chat Bot Pada Server Pulsa Menggunakan Telegram Bot API	Telegram Bot API, Long Polling	PHP, SQL	Server Isi Ulang Pulsa
Yoga Dwi Jatmika (2018)	Pembangunan Website E-Commerce pada UD. Karya Utama	Framework Codeigniter	PHP, SQL	Toko Bangunan
Usulan Penulis (2019)	Pembuatan Telegram Bot Menggunakan Metode Webhooks Untuk Pemesanan Rental Mobil (Studi	Telegram Bot API, Webhooks, Framework Codeigniter	PHP, SQL	Rental Mobil

Peneliti	Judul	Metode / Teknologi	Bahasa Pemrograman	Studi Kasus
	Kasus di Kuy Tour (Yogyakarta)			

2.2 Dasar Teori

2.2.1 Telegram Bot

Telegram mempersilahkan para pengembang untuk mengembangkan aplikasinya dengan Telegram API. Ada 2 (dua) jenis API yang disediakan Telegram, API yang pertama adalah klien Telegram dimana semua orang bebas untuk membuat, memodifikasi dan mendistribusikan aplikasi pesan instannya versi mereka sendiri. Untuk hal tersebut, disediakan *source code* yang digunakan pada saat ini sehingga pengembang tidak harus membangun aplikasi Telegram dari awal.

Jenis lainnya ialah Telegram Bot API, API jenis kedua ini memungkinkan pengembang untuk membuat Bot yang dapat membalas pesan dari semua penggunanya jika mengirimkan pesan perintah yang telah diatur dalam Bot itu sendiri. Layanan ini hanya tersedia bagi pengguna Telegram saja sehingga untuk dapat berkomunikasi dengan Bot Telegram, dibutuhkan aplikasi dan akun Telegram.

Telegram bot merupakan akun khusus yang tidak memerlukan nomor telepon tambahan sebagai syarat khususnya. Akun bot tersebut berfungsi sebagai antarmuka untuk kode yang dapat dijalankan pada server pengembang. Bot tersebut dapat melakukan beberapa pekerjaan yaitu:

- a. Mengintegrasikan dengan layanan lainnya

Bot dapat mengirimkan komentar jarak jauh atau mengendalikan smart home. Selain itu, bot juga mampu mengirimkan pemberitahuan melalui Telegram ketika terjadi sesuatu di suatu tempat.

- b. Menciptakan alat khusus

Bot mampu memberikan pemberitahuan maupun memberikan sebuah peringatan, ramalan cuaca, terjemahan, atau layanan lain.

- c. Membangun single player ataupun multiplayer game

Keunggulan lainnya yaitu bot mampu memainkan permainan seperti catur.

- d. Membangun layanan sosial

Sebuah bot dapat menghubungkan orang-orang untuk mencari mitra percakapan berdasarkan kepentingan bersama.

Dalam penggunaannya, pengembang tidak perlu repot untuk mengetahui protokol enkripsi Telegram karena hal tersebut akan ditangani oleh API Telegram. API Telegram berupa sebuah kode otentikasi yang disebut token. Token tersebut didapatkan ketika telah melakukan pendaftaran akun pada Telegram.

Pada implementasinya, pengembang hanya memerlukan token sebagai syarat untuk menggunakan Telegram bot. Pada Telegram bot API tersedia beberapa metode dalam pengiriman pesan yaitu `getMe`, `sendMessage`, `sendDocument`, `sendPhoto`, dan lain-lain (“All Method,” n.d.). Setiap metode tersebut harus memiliki parameter `chat_id` yang mendefinisikan identitas target obrolan.

Namun, terdapat perbedaan parameter pada setiap metode misalnya `sendMessage` wajib memiliki parameter `text` yang memiliki nilai berupa pesan yang

akan dikirim. Sedangkan `sendDocument` harus memiliki parameter `document` yang berisi file yang akan dikirimkan. Berikut daftar perintah yang akan dibuat pada Telegram bot :

- a. `/cara_registrasi`, untuk informasi cara mendaftar ke bot Telegram.
- b. `/cara_pesan`, untuk informasi cara pemesanan mobil melalui bot Telegram.
- c. `/daftar_mobil`, untuk melihat daftar mobil yang tersedia.
- d. `/konfirmasi_pemesanan`, untuk melakukan konfirmasi pembayaran dengan mengupload bukti transfer melalui bot Telegram.
- e. `/ubah_pemesanan`, untuk mengubah pesanan yang telah dibuat.
- f. `/batalkan_pemesanan`, untuk membatalkan pesanan yang telah dibuat.
- g. `/status_pemesanan`, untuk melihat detail status pemesanan yang telah dibuat (The Telegram Team, 2015).

2.2.2 Metode Webhook

Salah satu metode yang dapat digunakan untuk membuat Bot Telegram ialah webhook dengan `setWebhook`. Metode ini digunakan untuk menentukan url dan menerima pembaruan yang masuk melalui webhook keluar. Setiap kali ada pembaruan untuk bot, Telegram akan mengirimkan permintaan POST HTTPS ke url yang ditentukan, yang berisi pembaruan dengan bentuk JSON. Jika permintaan tidak berhasil, Telegram akan berhenti mengirim *request* setelah upaya yang wajar. Sebaliknya, respon *true* jika *request* dinyatakan sukses.

- h. Untuk memastikan bahwa permintaan Webhook adalah benar berasal dari Telegram, digunakan jalur rahasia pada URL, mis. `https://www.example.com/ <token>`. Karena token bot hanya diketahui oleh

pembuat bot itu sendiri, maka *request* dapat diyakini benar berasal dari Telegram (The Telegram Team, 2015).

Parameter yang dapat digunakan untuk membuat Bot Telegram dengan metode Webhook dapat dilihat pada tabel 2.2.

Tabel 2. 2 Parameter Telegram Bot Metode Webhook

Parameter	Tipe data	Dibutuhkan	Deskripsi
url	String	Ya	URL HTTPS untuk mengirim pembaruan ke bot.
Certificate	InputFile	Opsional	Digunakan untuk mengunggah sertifikat kunci publik sehingga sertifikat root yang digunakan dapat diperiksa.
max_connection	Integer	Opsional	Jumlah maksimum koneksi HTTPS simultan yang diizinkan ke webhook untuk pengiriman pembaruan, 1-100. Standarnya menjadi 40.
allowed_update	Array of String	Opsional	Daftar jenis pembaruan yang diinginkan untuk diterima bot. Jika tidak ditentukan, pengaturan sebelumnya akan digunakan.

2.2.3 REST API

Menurut Hafid (2015:12) REST adalah singkatan dari *REpresentational State Transfer*. Merupakan standar dalam arsitektur web yang menggunakan *Protocol* HTTP untuk pertukaran data. Konsep REST pertamakali diperkenalkan oleh Roy Fielding pada tahun 2000. Cara kerjanya, REST server menyediakan jalur untuk akses *resource* atau data, sedangkan REST client melakukan akses *resource* dan kemudian menampilkan atau menggunakannya. *Resource* yang dihasilkan sebenarnya berupa teks, namun formatnya bisa bermacam-macam tergantung keinginan developer, umumnya adalah JSON (*JavaScript Object Notation*) dan XML (*Extensible Markup Language*).

Dalam mengakses sebuah *resource*, REST juga menggunakan konsep URI dimana ada method yang digunakan, by default adalah GET. Berikut ini method-method yang mendukung REST:

- GET, cocok untuk *resource* yang hanya perlu dibaca saja (*read only*)
- PUT, cocok digunakan untuk membuat/*create resource* baru.
- DELETE, cocok digunakan untuk menghapus suatu *resource*.
- POST, cocok digunakan untuk meng-*update* suatu *resource*.
- OPTIONS, cocok digunakan untuk mendapatkan operasi yang didukung pada *resource*.

2.2.4 Web Server

Web server adalah perangkat lunak yang menyediakan layanan permintaan data dengan protokol http atau https yang diakses melalui aplikasi tertentu seperti web browser. Web server merespon permintaan dengan mengirimkan kembali konten berupa gambar, tulisan, atau lainnya dan kemudian ditampilkan melalui browser (Singh & Kumar, 2011).

2.2.5 PHP

PHP singkatan (PHP Hypertext Preprocessor) adalah bahasa server-side scripting yang menyatu dengan HTML untuk membuat halaman web yang dinamis. Server-side scripting adalah sintak dan perintah – perintah yang diberikan akan sepenuhnya dijalankan di server tapi disertakan pada dokumen HTML. (Bimo Sunarfrihantono, 2002).

2.2.6 MySQL

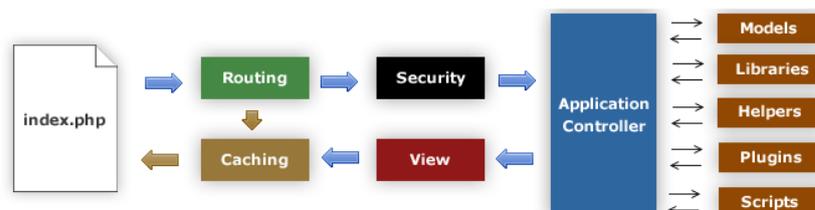
MySQL tergolong sebagai DBMS (Database Management System). Perangkat lunak ini bermanfaat untuk mengelola data yang sangat fleksibel dan juga cepat. MySQL banyak dipakai untuk penanganan database, selain handal juga bersifat opensource (Abdul Kadir, 2009).

2.2.7 Codeigniter

Codeigniter merupakan *open source* berbasis MVC (*Model View Controller*). Untuk membangun aplikasi web dinamis dengan cepat dan mudah. Codeigniter memiliki desain dan struktur file yang sederhana, didukung dengan dokumentasi yang lengkap sehingga *framework* ini lebih mudah dipelajari.

Codeigniter ini memungkinkan para pengembang untuk menggunakan *framework* secara parsial atau secara keseluruhan. Artinya bahwa codeigniter masih memberi kebebasan kepada para pengembang untuk menulis bagian – bagian kode tertentu di dalam aplikasi menggunakan cara konvensional atau dengan sintak umum di dalam PHP, tidak harus menggunakan aturan penulisan kode di Codeigniter (Septian, 2011).

Adapun alur dari aplikasi yang ditulis menggunakan Codeigniter adalah sebagai berikut:



Gambar 2. 1 Alur Penggunaan Codeigniter

File *index.php*, berfungsi sebagai *front controller*, menginisialisasi *resource* utama yang dibutuhkan untuk menjalankan Codeigniter.

1. *Router*, memeriksa *HTTP request* untuk menentukan apa yang harus dilakukan.
2. Jika file *cache* ada, dikirim langsung ke browser, melewati eksekusi sistem normal.
3. Keamanan, sebelum *controller* aplikasi dimuat, *HTTP request* dan setiap data pengguna yang dikirim dan disaring dahulu untuk keamanan.
4. *Controller* memuat model, *library* utama, *helper* dan setiap *resource* lainnya yang diperlukan untuk memproses permintaan khusus.
5. *View*, proses render kemudian dikirim ke web browser agar dapat dilihat. Jika caching diaktifkan, *view* dicache terlebih dahulu sehingga pada permintaan berikutnya dapat dilayani.

2.2.8 Unified Modeling Language (UML)

Menurut Suhendar dan Gunadi (2002), *Unified Modeling Language (UML)* adalah sebuah bahasa untuk menentukan, visualisasi, konstruksi dan mendokumentasikan *artifact* dari sistem *software*, untuk memodelkan bisnis dan sistem *nonsoftware* lainnya. UML merupakan suatu kumpulan teknik terbaik yang telah terbukti sukses dalam memodelkan sistem yang besar dan kompleks.

Artifact adalah sepotong informasi yang digunakan atau dihasilkan dalam suatu proses rekayasa perangkat lunak. *Artifact* dapat berupa model, deskripsi, atau perangkat lunak. Tujuan utama UML diantaranya untuk :

- a. Memberikan model yang siap pakai, bahasa permodelan visual yang ekspresif untuk mengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.
- b. Memberikan bahasa permodelan yang bebas dari berbagai bahasa pemrograman dan praktik rekayasa.
- c. Menyatukan praktik-praktik terbaik yang terdapat dalam permodelan.