

BAB 2

LANDASAN TEORI

2.1 Toko Grosir

Pedagang Grosir adalah perantara antara produsen dan pedagang ritel. Pedagang Grosir mendapatkan atau membeli produk dalam jumlah besar dengan harga yang lebih rendah dari produsen dan menjualnya ke pedagang eceran atau perusahaan ritel. Pengecer yang membeli barang dari Grosir tersebut akan menjual kembali ke konsumen akhir atau untuk keperluan bisnis lainnya. Pedagang Grosir dapat juga dikatakan sebagai pedagang atau orang yang usahanya membeli barang dalam jumlah besar dan menjualnya dalam jumlah yang lebih kecil, misalnya ke pedagang toko atau ke pengecer (Utami, 2021).

2.2 Android

Android adalah sebuah sistem operasi yang sudah menjadi salah satu kebutuhan pokok, di mana sistem Android ini berbasis Linux. Hampir semua orang membutuhkan alat yang satu ini untuk berkomunikasi, mencari informasi dan perkembangan berita terbaru, melakukan pekerjaan, dan berbagai aktivitas lainnya. Bahkan, saat ini hampir semua orang menggantungkan aktivitas mereka pada android yang mereka miliki, dari penggunaan *social media*, *entertainment*, berita, pemesanan makanan, dan lainnya. Android juga memiliki berbagai macam fitur antara lain (Enterprise, 2015).

Tabel 2.1. Fitur - fitur Android

Fitur	Deskripsi
Antarmuka	Layar sistem operasi Android menampilkan antarmuka yang bagus dan intuitif.
Konektifitas	GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE, NFC dan WiMAX.
Penyimpanan	Penyimpanan SQLite, database relasional ringan yang digunakan untuk penyimpanan data.
Media	H.263, H.264, MPEG-4 SP, AMR, AMR-WB, AAC, HE-AAC, AAC 5.1, MP3, MIDI, Ogg Vorbis, WAV, JPEG, PNG, GIF, dan BMP.
Messaging	SMS dan MMS
Web browser	Berdasarkan layout engine WebKit, dengan Chrome's V8 JavaScript Engine yang mendukung HTML5 dan CSS3.
Multi-touch	Android memiliki dukungan multi-touch (lebih dari satu sentuhan layar).
Multi-tasking	Pengguna dapat berpindah dari satu task ke task lainnya dan beberapa aplikasi dapat berjalan pada waktu
Multi-language	Mendukung teks dalam berbagai bahasa
Wi-Fi Direct	ologi yang memungkinkan aplikasi untuk menemukan dan memasang secara langsung, melalui koneksi peer-to-peer high-bandwidth

(Enterprise, 2015)

2.3 MySQL

MySQL adalah sebuah DBMS (Database Management System) yang menggunakan SQL (Structured Query Language) biasanya banyak digunakan untuk pembuatan aplikasi berbasis website. MySQL memiliki dua lisensi, pertama adalah *Free Software* di mana perangkat dapat diakses siapa saja. Dan kedua adalah *Shareware*, di mana perangkat lunak berpemilik mempunyai batasan untuk penggunaannya.

MySQL juga termasuk sebagai RDBMS (Relational Database Management System). Sehingga, menggunakan tabel, kolom, baris, di dalam struktur pembuatan databasenya. karena itu dalam proses pengambilan data menggunakan metode relational database. Dan juga dapat berperan menjadi penghubung antara perangkat lunak ke database server.

MySQL memiliki beberapa fungsi antara lain adalah untuk membuat dan mengelola database server yang memuat segala jenis informasi dengan menggunakan SQL. Fungsi lainnya adalah memudahkan pengguna untuk mengakses data berisi informasi yang berbentuk String (teks), di mana data tersebut dapat diakses secara personal ataupun publik dalam web.

Antarmuka dari MySQL adalah PHPMyAdmin. Di mana berfungsi untuk menghubungkan macam - macam bahasa pemrograman PHP dengan MySQL untuk proses pengelolaan basis data web, Mysql juga memiliki beberapa kelebihan diantaranya(Adani, 2020).

1. Bersifat Open Source

Penggunaan dari MySQL dapat diakses oleh siapapun dan kapanpun. Karena memiliki dua lisensi, berupa *Free Software* dan *Shareware*. Selain itu juga termasuk dalam sistem manajemen database yang dapat digunakan secara gratis. Dan, memiliki banyak komunitas beserta dokumentasi untuk membantu dalam mengembangkan aplikasi yang menggunakan database server.

2. Mendukung Penggunaan Multi User

MySQL juga dapat digunakan secara bersamaan dalam satu waktu dikarenakan bersifat Multi User. Sehingga, tidak perlu khawatir saat mengakses data secara bersamaan.

3. Terintegrasi dengan Bahasa Pemrograman yang lain

Dapat terintegrasi dengan bahasa pemrograman yang lain seperti pada PHP. Dengan menggunakan PHPMyAdmin sebagai perantara, developer sudah bisa membuat data pada web server dengan efektif. Kemudian, dapat digunakan untuk keperluan pengembangan perangkat lunak dengan menggunakan framework seperti CodeIgniter dan Laravel.

4. Struktur Tabel Cenderung Fleksibel

Jika dibandingkan dengan Oracle dan Postgre, MySQL tergolong lebih fleksibel. Karena sebagai contoh dalam proses eksekusi pada ALTER TABLE lebih mudah digunakan dan cepat diproses.

5. Sistem Keamanan yang Telah Terjamin

MySQL dapat mengatur akses pengguna (user) dengan enkripsi. Selain itu, terdapat beberapa fitur keamanan yang lain seperti, level nama pada host.

2.4 Code Igniter

CodeIgniter adalah web application network yang bersifat open source dan digunakan untuk membangun aplikasi php secara dinamis. CodeIgniter adalah sebuah framework PHP dengan model MVC (Model, View, Controller) yang dapat membangun website dinamis dengan menggunakan PHP dan dapat mempercepat pengembang untuk membuat aplikasi web. Selain cepat dan ringan, CodeIgniter juga memiliki dokumentasi yang lengkap dan disertai contoh implementasi kodanya.

Dokumentasi yang lengkap inilah yang menjadi alasan kuat banyaknya orang memilih CodeIgniter sebagai framework pilihannya. Karena kelebihan yang dimiliki CodeIgniter, pembuat PHP Rasmus Lerdorf memuji CodeIgniter di frOSCon dengan mengatakan bahwa dia menyukai CodeIgniter karena “it is faster, lighter and the least like a framework.”

CodeIgniter pertamakali dikembangkan pada tahun 2006 oleh Rick Ellis. Dengan logo api yang menyala, CodeIgniter mengembangkan web dinamis dengan cepat dan mudah, berikut beberapa keunggulan menggunakan CodeIgniter(Admin, 2020).

1. Size yang kecil

Size framework Code igniter hanya sebesar 2MB. Size sekecil ini dapat memudahkan kita untuk menggunakan Code igniter, menjalankannya, juga meng-update.

2. Sangat cepat

Code igniter memiliki daya respon yang sangat cepat, di mana loading untuk menjalankannya kurang dari 50 milisecond.

3. Fleksibel

Code igniter memiliki komponen dan fitur yang berdiri sendiri tanpa terlalu bergantung pada komponen lainnya, membuat Code igniter sangat mudah untuk di-upgrade dan dicari fixnya jika ada masalah.

4. Arsitektur yang bersifat MVC (Model-View-Controller)

Ketika membuat atau mengerjakan aplikasi yang berbasis web, desain arsitektur MVC merupakan hal standar industri untuk sebuah framework. MVC sangat berguna dalam memisahkan data, business logic, dan presentasi.

5. Dapat dimodifikasi

Codeigniter sudah terdapat library dan bantuan dari awal mula pemakaian. Jika apa yang developer cari tidak ada, developer dapat dengan mudah menambahkan library sendiri.

6. Mudah dipelajari dan dikuasai

Codeigniter adalah framework yang sangat mudah untuk dipelajari. Dalam waktu yang singkat, developer dapat menciptakan dan mengembangkan aplikasi menggunakan Codeigniter.

2.5 Push Notification

Push notification adalah pesan notifikasi yang muncul di layar *device* seperti notifikasi di laptop, *handphone*, dan juga browser. Ia akan memberitahukan jika ada postingan terbaru pada sebuah website tersebut (Setiawan dan Kristianto, 2015). *Push notification* juga terbagi menjadi 2 yaitu :

1. *In-app push notification*

(a) *In-app push notification* adalah notifikasi yang Anda terima di *handphone* atau *tablet* Anda. *In-app push notification* biasa dikirimkan dari aplikasi yang Anda install dan kirim. Misalnya, kalau Anda mendapat notifikasi dari Facebook saat Anda tidak sedang membuka aplikasi Facebook adalah contoh *push notification*. Biasanya di *in-app push notification*, mereka akan memasukkan link agar Anda masuk kembali ke aplikasi.

2. *Web push notification*

(a) *Web push notifications* adalah notifikasi yang muncul dari website dan bukan aplikasi. *Push notification* yang ada di browser biasa muncul ketika pengguna mengunjungi website untuk pertama kalinya. Pengguna tidak perlu membuat aplikasi dan tinggal memasang notifikasi

itu. Anda bisa mengirimkan push notifications ke Safari, Chrome, and Firefox. Dengan web *push notifications* pengguna bisa menjangkau pengunjung website pengguna dan kirimkan pesan yang bisa menarik website Anda. Jadi meskipun pengunjung website tidak berlangganan ke website pengguna, mereka tetap memiliki hubungan dengan pengguna(Dewaweb, 2021).

2.6 Firebase Cloud Messaging

Firebase Cloud Messaging adalah fitur yang terdapat pada Firebase di mana fitur ini berfungsi untuk melakukan pertukaran pesan yang dapat digunakan untuk mengirim pesan atau notifikasi secara online. Dengan Firebase Cloud Messaging, satu aplikasi dapat memberi tahu aplikasi lain untuk bertukar informasi atau pengguna dapat mengirim pesan notifikasi untuk mendorong interaksi kembali dan retensi pengguna lainnya

Firebase Cloud Messaging memiliki beberapa kemampuan. Kemampuan utama dari firebase Cloud Messaging sebagai berikut(FirebaseGoogle, 2020).

1. Mengirim pesan notifikasi atau pesan data

- Mengirim pesan notifikasi yang ditampilkan kepada pengguna. Atau mengirim pesan data dan menentukan sepenuhnya apa yang terjadi dalam kode aplikasi. Terdapat 2 jenis pesan yaitu Pesan notifikasi, di mana pesan notifikasi ini terkadang dianggap sebagai ”pesan tampilan”. Pesan ini ditangani oleh Firebase Cloud Messaging (FCM) SDK secara otomatis. Lalu jenis yang lainnya adalah pesan data, di mana pesan data ditangani langsung oleh aplikasi klien.

2. Penargetan pesan serbaguna

- Mendistribusikan pesan ke aplikasi klien dengan salah satu dari 3 cara

yaitu mendistribusikan pesan ke sebuah perangkat, ke sebuah kumpulan grup perangkat, atau ke perangkat yang sedang berlangganan.

3. Mengirim pesan dari aplikasi klien

- Mengirim konfirmasi, chat, dan pesan lain dari perangkat kembali ke server melalui saluran koneksi FCM yang andal dan hemat baterai.

2.7 BlackBox Testing

Black Box Testing (Pengujian Kotak Hitam), juga disebut pengujian perilaku, berfokus pada persyaratan fungsional perangkat lunak. Di mana, teknik *Black Box Testing* memungkinkan pengguna untuk membuat beberapa kumpulan kondisi masukan yang sepenuhnya akan melakukan semua kebutuhan untuk program yang digunakan. *Black box testing* ini bukan teknik alternative untuk *Black box*. Sebaliknya, ini merupakan pendekatan pelengkap yang mungkin dilakukan untuk mengungkap kelas kesalahan yang berbeda dari yang diungkap oleh metode kotak putih (Anggun Nugroho, 2018).

Metode *Black box Testing* merupakan salah satu metode yang mudah digunakan karena hanya memerlukan batas bawah dan batas atas dari data yang diharapkan. Estimasi banyaknya data uji dapat dihitung melalui field data entri yang diuji, aturan entri yang harus dipenuhi serta batas atas dan batas bawah yang memenuhi. Dan dengan metode ini dapat diketahui jika fungsionalitas masih dapat menerima masukan data yang tidak diharapkan maka menyebabkan data yang disimpan kurang valid.

Pengujian metode *black box testing* adalah satu set aktifitas yang direncanakan dan sistematis untuk menguji atau mengevaluasi kebenaran yang diinginkan. Aktifitas pengujian terdiri dari satu set atau sekumpulan langkah di mana dapat menempatkan desain kasus uji yang spesifik. Kualitas perangkat lunak bergantung kepada kepuasan pelanggan dan kualitas sebuah perangkat lunak perlu

dijaga(Cholifah et al., 2018).

Didalam metode *black box testing* terdapat 10 tipe pengujian, diantaranya(Hanifah et al., 2016).

1. *Equivalence Partitioning*

Membagi inputan menjadi kelas data yang dapat digunakan untuk menggenerasi kasus uji.

2. *Boundary Value Analysis / Limit Testing*

Mengijinkan untuk menyeleksi kasus uji yang menguji batasan nilai input, Merupakan komplemen dari *Equivalence Partitioning*.

3. *Comparison Testing*

Uji setiap versi dengan data yang sama untuk memastikan semua versi menghasilkan keluaran yang sama.

4. *Sample Testing*

Melibatkan beberapa nilai yang terpilih dari sebuah kelas ekivalen.

5. *Robustness Testing*

Data input dipilih diluar spesifikasi yang telah didefinisikan, Tujuan dari pengujian ini adalah membuktikan bahwa tidak ada kesalahan jika masukan tidak valid

6. *Behavior Testing*

Hasil uji tidak dapat dievaluasi jika hanya melakukan pengujian sekali, tapi dapat dievaluasi jika pengujian dilakukan beberapa kali, misalnya pada pengujian struktur data stack.

7. *Performance Testing*

Mengevaluasi kemampuan program untuk beroperasi dengan benar dipandang dari sisi acuan kebutuhan misalnya : aliran data, ukuran pemakaian memori, kecepatan eksekusi

8. *Requirement Testing*

Spesifikasi kebutuhan yang terasosiasi dengan perangkat lunak diidentifikasi pada tahap spesifikasi kebutuhan dan desain

9. *Endurance Testing*

Melibatkan kasus uji yang diulang-ulang dengan jumlah tertentu.

10. *Cause – Effect Relationship Testing*

Bagi-bagi spesifikasi kebutuhan menjadi bagian yang memiliki kemungkinan kerja