

BAB II

LANDASAN TEORI

2.1 Penjualan

Penjualan adalah suatu usaha yang terpadu untuk mengembangkan rencana-rencana strategis yang diarahkan pada usaha pemuasan kebutuhan dan keinginan pembeli, guna mendapatkan penjualan yang menghasilkan laba (Nurchayono, F. 2017) yang dikutip dari (Marwan, 1991:30). Penjualan merupakan sumber hidup suatu perusahaan, karena dari penjualan dapat diperoleh laba serta suatu usaha memikat konsumen yang diusahakan untuk mengetahui daya tarik mereka sehingga dapat mengetahui hasil produk yang dihasilkan. Menurut Winardi (1982:89) yang dikutip (Nurchayono, F. 2017), penjualan adalah suatu transfer hak atas benda-benda. Dari penjelasan tersebut dalam memindahkan atau mentransfer barang dan jasa diperlukan orang-orang yang bekerja dibidang penjualan seperti pelaksanaan dagang, agen, wakil pelayanan dan wakil pemasaran. Menurut Joel G. Siegel dan Joe K. Shim (2009:67) yang dikutip (Nurchayono, F. 2017): “Penjualan adalah Penerimaan yang diperoleh dari pengiriman barang dagangan atau dari penyerahan pelayanan dalam bursa sebagai barang pertimbangan. Pertimbangan ini dapat dalam bentuk tunai peralatan kas atau harta lainnya. Pendapatan dapat diperoleh pada saat penjualan, karena terjadi pertukaran, harga jual dapat ditetapkan dan bebannya diketahui”. (Nurchayono, F, 2017).

2.2 Aplikasi Mobile

Aplikasi *Mobile* adalah perangkat lunak yang berjalan pada perangkat *mobile* seperti *smartphone* atau *tablet PC*. Aplikasi *Mobile* juga dikenal sebagai aplikasi yang dapat diunduh dan memiliki fungsi tertentu sehingga menambah fungsionalitas dari perangkat *mobile* itu sendiri. Untuk mendapatkan *mobile application* yang diinginkan, *user* dapat mengunduhnya melalui situs tertentu sesuai dengan sistem operasi yang dimiliki. Google Play dan iTunes merupakan beberapa contoh dari situs yang menyediakan beragam aplikasi bagi pengguna

Android dan iOS untuk mengunduh aplikasi yang diinginkan. (Musa, H., Chung, S. H. L., Abal Abas, Z., & Mohamad, N. 2016) yang dikutip dari (*Mobile Marketing Association*, 2015).

Berdasarkan jenisnya, (Dikanata, D., & Wati, R. 2018) yang dikutip dari Brian Fling (2009:70) membagi aplikasi mobile menjadi beberapa kelompok yaitu:

a. *Short Message Service (SMS)*

Merupakan aplikasi mobile paling sederhana, dirancang untuk ber kirim pesan dan berguna ketika terintegrasi dengan jenis aplikasi *mobile* lainnya.

b. *Mobile Websites (Situs Web Mobile)*

Merupakan situs *web* yang dirancang khusus untuk perangkat *mobile*. Situs *web mobile* sering memiliki desain yang sederhana dan biasanya bersifat memberikan informasi.

c. *Mobile Web Application (Aplikasi Web Mobile)* Aplikasi *web mobile* merupakan aplikasi *mobile* yang tidak perlu diinstal atau dikompilasi pada perangkat target. Menggunakan XHTML, CSS, dan JavaScript, aplikasi ini mampu memberikan pengguna pengalaman layaknya aplikasi *native/asli*.

d. *Native Application (Aplikasi Asli)*

Merupakan aplikasi mobile yang harus diinstal pada perangkat target. Aplikasi ini dapat disebut aplikasi *platform*, karena aplikasi ini harus dikembangkan dan disusun untuk setiap *platform mobile* secara khusus.

2.3 Android

Android adalah sebuah sistem operasi untuk smartphone dan Tablet. Sistem operasi dapat diilustrasikan sebagai ‘jembatan’ antara piranti (*device*) dan penggunanya, sehingga pengguna bisa berinteraksi dengan *device*-nya dan menjalankan aplikasi-aplikasi yang tersedia pada *device* (Yuni Puspita Sari, 2016).



Gambar 2.1 Logo Android Studio

Android Studio adalah sebuah IDE (*Integrated Development Environment*) untuk Android development yang diperkenalkan oleh Google. Android Studio merupakan pengembangan dari Eclipse IDE, dan dibuat berdasarkan IDE Java populer, yaitu IntelliJ IDEA. Android studio mempunyai fitur-fitur yang dapat mempermudah penulis dalam penelitiannya.

2.3.1 Elemen Android

1. Dalvik Virtual Machine (DVM)

Salah satu element kunci dari Android adalah Dalvik Virtual Machine (DVM). Android berjalan di Dalvik Virtual Machine (DVM) bukan di Java Virtual Machine (JVM), sebenarnya banyak persamaannya dengan Java Virtual Machine (JVM) seperti Java ME (Java Mobile Edition), tetapi Android menggunakan virtual machine sendiri yang dirancang untuk memastikan beberapa fitur-fitur berjalan lebih efisien pada perangkat mobile.

2. Android SDK (Software Development Kit)

Android SDK adalah tools API (Application Programming Interface) yang diperlukan untuk mulai mengembangkan aplikasi pada *platform* Android menggunakan bahasa pemrograman Java. Android merupakan subset perangkat lunak untuk ponsel yang meliputi sistem operasi middleware dan aplikasi kunci yang di-release oleh Google. Saat ini disediakan Android SDK (Software Development Kit) sebagai alat bantu dan API untuk mulai mengembangkan aplikasi pada *platform* Android menggunakan bahasa pemrograman Java. Sebagai *platform* aplikasi-netral, Android memberi kesempatan untuk membuat aplikasi yang dibutuhkan.

3. ADT (Android Development Tools)

Android development tools adalah plugin yang di desain untuk IDE Android Studio yang memberikan kemudahan dalam mengembangkan aplikasi Android dengan menggunakan IDE Android Studio. Dengan menggunakan ADT untuk Android Studio akan memudahkan dalam membuat aplikasi project Android, membuat GUI aplikasi, dan menambahkan komponen-komponen yang lainnya.

2.4 Perangkat Lunak Yang Digunakan

2.4.1 JAVA

Java adalah bahasa pemrograman yang *multi device*, artinya Sekali menuliskan sebuah program dengan menggunakan Java, maka dapat menjalankannya hampir di semua komputer dan perangkat lain yang support Java, dengan sedikit perubahan atau tanpa perubahan sama sekali dalam kodenya. Java adalah Bahasa yang dapat dijalankan disembarang *platform*, di beragam lingkungan internet, *consumer electronic products*, dan *computer applications*. Jadi berdasarkan pengertian diatas Java merupakan pemrograman yang dapat dijalankan multi *platform* (bisa berjalan di berbagai macam sistem operasi) yang mendukung Java dan menjalankan perangkat lunak ataupun jaringan (Afrizal 2017:3).

2.4.2 HTML (*Hyper Text Markup Language*)

Hypertext Markup Language (HTML) adalah bahasa pemrograman dengan format standar untuk membuat dokumen web yang juga merupakan Bahasa Markup bertanda, yang menggunakan rangkian text tertentu berupa kode-kode (Tag) yang dimengerti oleh web browser dan dapat menampilkannya di layar monitor, serta untuk menandai text yang mempunyai interpretasi khusus, HTML sendiri merupakan subset dari *Standard Generalized Markup Language* (SGML) yang pada perkembangannya kini HTML merupakan standar internet yang didefinisikan (Setiani, A. 2019) yang dikutip dari (Hajar dan Nasroh 2018).

2.4.3 Android Studio

Android studio adalah IDE (*Integrated Development Environment*) resmi untuk pengembangan aplikasi Android dan bersifat *open source* atau gratis. Peluncuran Android Studio ini diumumkan oleh Google pada 16 mei 2013 pada *event* Google I/O *Conference* untuk tahun 2013. Sejak saat itu, Android Studio menggantikan Eclipse sebagai IDE resmi untuk mengembangkan aplikasi Android Android studio sendiri dikembangkan berdasarkan IntelliJ IDEA yang mirip dengan Eclipse disertai dengan (Android Development Tools) ADT plugin (Kurniawan, A., Yosman, D., Arif, A., & Juansah, J. 2015).

Fitur yang dimiliki oleh Android studio:

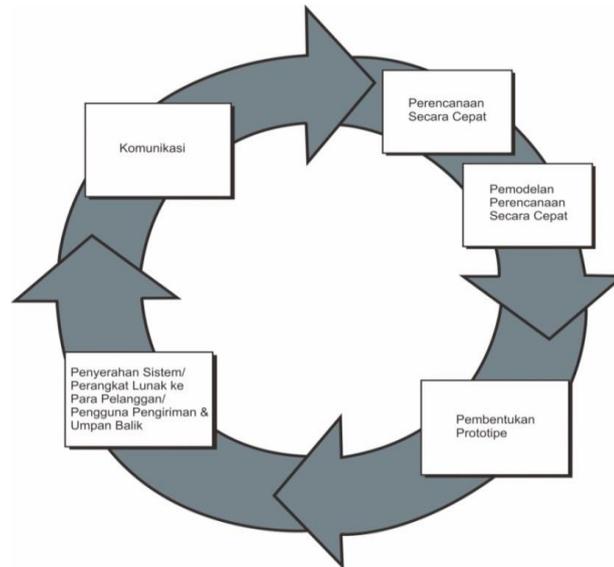
- a. Proyek berbasis pada *Gradle Build*.
- b. *Refactory* dan pembenahan bug yang cepat.
- c. Tools baru yang bernama “*Lint*” dikalim dapat memonitor kecepatan, kegunaan, serta kompetibelitas aplikasi dengan cepat.
- d. Mendukung *Proguard And App-signing* untuk keamanan.
- e. Memiliki GUI aplikasi android lebih mudah.
- f. Didukung oleh *Google Cloud Platfrom* untuk setiap aplikasi yang dikembangkan.

2.4.4 XAMPP

XAMPP adalah perangkat lunak bebas, yang mendukung lebih dari satu sistem operasi, merupakan kumpulan dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (localhost), yang terdiri atas program Apache http Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP. Program ini tersedia dalam GNU (*General Public License*) merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis (Laisina, F. M., Mandagi, C. K., & Maramis, F. R. 2019).

2.5 Metode Prangkat Lunak Yang Digunakan

Metode yang digunakan adalah metode *prototype*, *prototype* merupakan metode yang efektif dalam merancang perangkat lunak. *Prototype* dimulai dengan mengumpulkan kebutuhan yang akan di rancang. Pengembang mendefinisikan *object* keseluruhan dari perangkat lunak, mengidentifikasi segala aktifitas yang diketahui dan kemudian melakukan “perancangan kilat”. Perancangan kilat berfokus pada penyajian dari aspek-aspek perangkat lunak tersebut yang akan nampak bagi pelanggan atau pemakai (contohnya pendekatan *input* dan format *output*), *prototype* memiliki 5 tahapan seperti pada gambar 2.2 berikut:



Gambar 2.2 Metode *Prototype*

2.6 Pengujian *Black-Box Testing*

Metode pengujian pada aplikasi ini menggunakan *Black Box Testing* yaitu yang berfokus pada spesifikasi fungsional dari perangkat lunak. Tester dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program.

Black Box Testing bukanlah solusi alternatif dari *White Box Testing* tapi lebih merupakan pelengkap untuk menguji hal-hal yang tidak dicakup oleh *White Box* (Testing Hertanto, D. B., Nugroho, S., & Prihatanta, H. 2018) yang dikutip dari (Pressman 2012). *Black Box Testing* cenderung untuk menemukan hal-hal berikut:

1. 1.Fungsi yang tidak benar atau tidak ada.
2. 2.Kesalahan antarmuka (*interface errors*).
3. 3.Kesalahan pada struktur data dan akses basis data.
4. 4.Kesalahan performansi (*performance errors*).
5. 5.Kesalahan inisialisasi dan terminasi.

2.7 Sistem Pemodelan

2.7.1 *Unified Modelling Language (UML)*

Unified Modeling Language (UML) adalah sebuah bahasa yang berdasarkan gambar untuk memvisualisasikan, menspesifikasikan, membangun dan pendokumentasian dari sebuah sistem pengembangan perangkat lunak berbasis

Objek. *Unified Modeling Language* (UML) bukanlah merupakan bahasa pemrograman tetapi model-model yang tercipta berhubungan langsung dengan berbagai macam bahasa pemrograman berorientasi obyek, seperti Java (Syafitri 2016). UML tersusun atas sejumlah elemen grafis membentuk diagram-diagram. Dalam penelitian ini melakukan desain hanya 2 diagram yaitu *Use Case* Diagram dan *Activity* Diagram.

2.7.2 Use Case Diagram

Diagram ini menunjukkan fungsionalitas suatu sistem atau kelas dan bagaimana sistem ini berinteraksi dengan dunia luar. *Use case* diagram dapat digunakan untuk memperoleh kebutuhan sistem dan memahami bagaimana sistem seharusnya bekerja (Syafitri, Y. 2016).

Berikut ini adalah simbol-simbol yang ada pada *Use Case* Diagram terdapat pada tabel 2.1

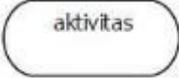
Tabel 2.1 Simbol *Use Case* Diagram

SIMBOL	NAMA	KETERANGAN
	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.
	<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
	<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
	<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
	<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (<i>sinergi</i>).
	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

2.7.3 Activity Diagram

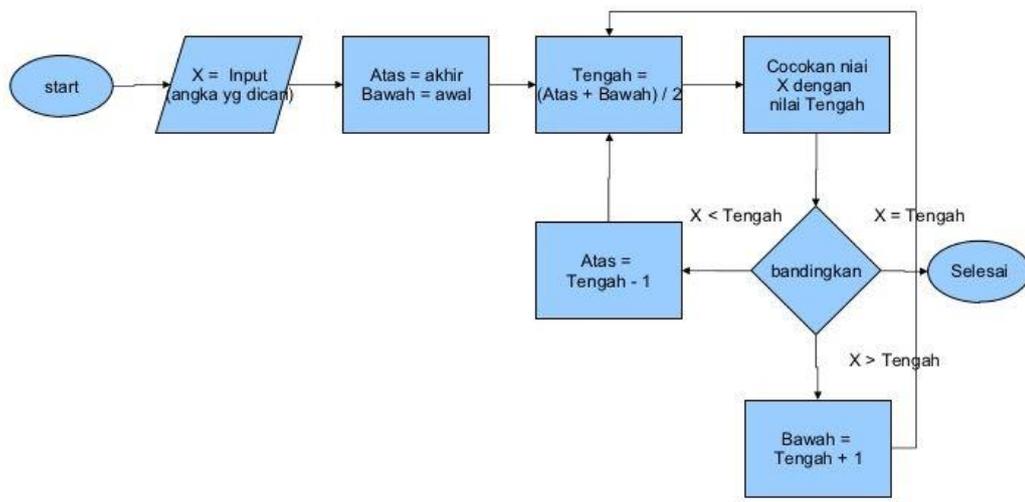
Activity Diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, keputusan yang mungkin terjadi, dan bagaimana suatu aktivitas berakhir. *Activity* diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa kegiatan. Aktivitas menggambarkan proses yang akan berjalan, sedangkan *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas (Syafitri, Y. 2016). Berikut ini adalah simbol-simbol yang ada pada *Activity* Diagram terdapat pada tabel 2.2

Tabel 2.2 Simbol *Activity* Diagram

Simbol	Deskripsi
status awal 	status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
aktivitas 	aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
percabangan / <i>decision</i> 	asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
penggabungan / <i>join</i> 	asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
status akhir 	status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
Database 	Digunakan untuk menunjukkan Penyimpanan pada database sistem
<i>fork</i> , 	digunakan utk menunjukkan kegiatan yg dilakukan secara paralel
<i>join</i> , 	digunakan utk menunjukkan kegiatan yg digabungkan

2.8 Binary Search

Binary Search (pencarian biner) dapat dilakukan jika data sudah dalam keadaan urut. Dengan kata lain, apabila data belum dalam keadaan urut, pencarian biner tidak dapat dilakukan. Dalam kehidupan sehari-hari, sebenarnya kita juga sering menggunakan pencarian biner. Misalnya saat ingin mencari suatu kata dalam kamus. Adapun flowchart pencarian *binary* seperti pada gambar 2.3 berikut:



Gambar 2.3 Flowchart Binary Search

Prinsip dari pencarian biner dapat dijelaskan sebagai berikut :

1. Mula-mula diambil posisi awal = 1 dan posisi akhir = N
2. Cari posisi data tengah dengan rumus (posisi awal + posisi akhir) / 2.
3. Data yang dicari dibandingkan dengan data tengah.
4. Jika lebih kecil, proses dilakukan kembali tetapi posisi akhir dianggap sama dengan posisi tengah-1.
5. Jika lebih besar, proses dilakukan kembali tetapi posisi awal dianggap sama dengan posisi tengah + 1.
6. Demikian seterusnya sampai data tengah sama dengan yang dicari.

2.8.1 Pohon (*Tree*)

Dalam kehidupan sehari-hari kita telah akrab dengan bentuk struktur pohon, khususnya menggambarkan sesuatu yang berjenjang/berhirarki. Sebuah pohon (*tree*) adalah himpunan berhingga yang beranggotakan satu atau lebih data/node,

sedemikian hingga dapat ditentukan sebuah node khusus terpartisi atau terpisah menjadi beberapa sub himpunan yang masing-masing juga sebuah pohon yang disebut subpohon/anak pohon (*subtree*).

2.8.2 Pohon Biner (*Binary Tree*)

Menurut Heri Sismoro sebuah pohon biner (*binary tree*) adalah sebuah himpunan berhingga atas data-data atau node-node, baik yang berupa himpunan kosong ataupun yang tidak kosong sedemikian hingga darinya dapat ditetapkan sebuah elemen $t \in \text{binary tree dan bynery tree} - \{t\}$ terpartisi menjadi dua buah sub himpunan pohon biner 1, dan pohon biner 2, yang keduanya juga merupakan pohon biner (*binary tree*) dan disebut subpohon kiri dan subpohon biner kanan (*left and right sub binary tree*). Kedua subpohon ini juga biasa disebut anak kiri dan anak kanan.

Pohon atau *tree* adalah sebuah pengorganisasian secara hirarki dari beberapa buah simpul, dimana masing-masing simpul tidak mempunyai anak lebih dari dua. Simpul yang berada dibawah sebuah simpul dinamakan anak (*child*) dari simpul tersebut. Simpul yang berada diatas sebuah simpul dinamakan induk (*parent*) dari simpul tersebut. (Syahputra, G. 2016)

2.9 Penelitian Terkait

Dalam penyusunan skripsi ini, penulis sedikit banyak terinspirasi dan mereferensi dari penelitian-penelitian sebelumnya yang berkaitan dengan latar belakang masalah pada skripsi ini.

1. Pembuatan Sistem Informasi Penjualan Pada Toko Sehat Jaya Elektronik Pacitan (Indah, I. N. 2013).

Sistem Informasi merupakan peranan sangat penting dalam kegiatan bisnis di suatu perusahaan. Toko Sehat Jaya Elektronik masih menggunakan cara yang konvensional dalam mengolah data tentang transaksi penjualan, untuk itu membutuhkan sistem informasi yang dapat menunjang kelancaran dalam melakukan transaksi penjualan. Tujuan dari penelitian ini merupakan upaya untuk membuat sistem informasi di Toko Sehat Jaya Elektronik yang belum

terkomputerisasi dan masih menggunakan cara yang konvensional. Dari penelitian ini diharapkan pembuatan sistem informasi yang dihasilkan nantinya akan mampu mengatasi dari sistem yang belum terkomputerisasi.

2. Penyusunan strategi dan sistem penjualan dalam rangka meningkatkan penjualan Toko Damai (Widharta, W. P. 2013)

Persaingan dalam dunia bisnis dewasa ini sangatlah ketat seiring dengan semakin meningkat dan berkembangnya dunia bisnis modern. Dengan bertambahnya jumlah pelaku bisnis yang bertujuan untuk memenuhi kebutuhan konsumen baik berupa barang atau jasa, para pelaku bisnis memiliki tantangan untuk mendapatkan pangsa pasar yang ada. Pada penelitian ini, peneliti mencoba meneliti tentang pengaruh variabel bauran pemasaran terhadap peningkatan penjualan Toko Damai yang bergerak dalam bidang menjual bahan – bahan bangunan. Adapun yang menjadi masalah dalam penelitian ini adalah untuk mengetahui seberapa besar pengaruh bauran pemasaran dalam meningkatkan penjualan Toko Damai.

3. Sistem Pencarian Obat pada Apotek (Yulia, R., & Susandri, S. 2018).

Apotek merupakan sebuah usaha yang menangani peracikan dan penjualan obat yang banyak tersedia di setiap daerah, salah satunya di Riau. Hingga saat ini untuk mendapatkan informasi keberadaan obat yang dibeli masyarakat harus mengunjungi ke setiap apotek dimana akan memerlukan waktu dan biaya tambahan untuk mencarinya. Oleh karena itu perlu dibangunnya sistem pencarian obat pada apotek dengan memfasilitasi pengguna untuk mendapatkan informasi apotek yang menjual obat berdasarkan kata kunci nama obat, jenis obat dan nama apotek. Metode pengembangan sistem yang digunakan penelitian ini yaitu waterfall. Dengan adanya sistem tersebut dapat membantu masyarakat dalam mencari obat-obatan yang dibutuhkan tanpa mendatangi apotek langsung.

4. Implementasi Teknik *Binary Search* Pada Kamus Indonesia batak Toba (Syahputra, G. 2016).

Pencarian (*searching*) merupakan pekerjaan yang sering kita lakukan dalam kehidupan sehari – hari. Dalam *text editor*, kita sering melakukan pekerjaan

mencari kata, atau mencari kata 1 dan menggantikannya dengan kata 2. Metode pencarian (*searching*) terbagi lima, yaitu metode pencarian linier (*Linear / Sequential Search*), metode pencarian biner (*Binary Search*), metode pencarian interpolasi (*Interpolation Search*), metode pencarian langsung (*Direct Search*), metode pencarian relatif (*Hash Search*). Masing-masing metode memiliki kelebihan dan kekurangan. Salah satu metode pencarian (*searching*) yang memiliki efisiensi adalah metoda pencarian biner (*Binary Search*). Pencarian biner (*Binary Search*) dapat dilakukan pada barisan bilangan yang telah diurutkan, baik secara menaik (*ascending*) maupun secara menurun (*discending*). Dari uraian diatas penulis tertarik menerapkan metode pencarian biner (*binary search*) pada kamus bahasa Indonesia – Batak toba, yang bertujuan untuk mengatasi kendala komunikasi yang sering terjadi, yang dapat menghambat untuk berinteraksi serta berkomunikasi dengan suku dari daerah lain. Disamping itu juga untuk menambah minat masyarakat mempelajari bahasa batak Toba terutama orang batak yang tinggal menetap di perantauan, dan juga untuk mengurangi kesalahan yang sering terjadi dalam proses penerjemahan, dalam hal ini khususnya adalah penerjemahan kata dari bahasa Indonesia kedalam kata bahasa Batak.