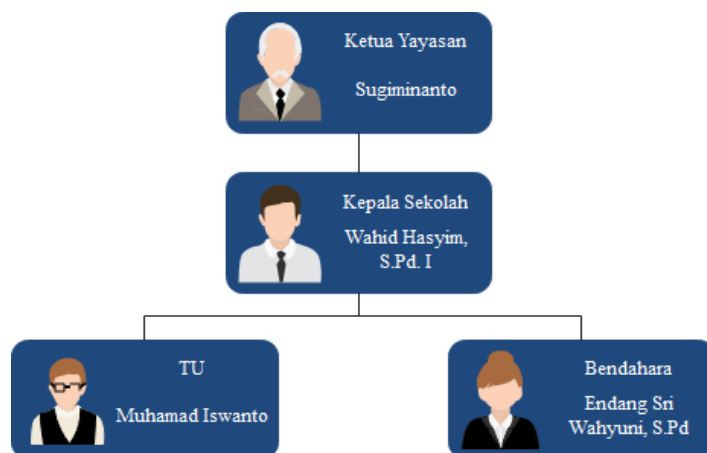


BAB II TINJAUAN PUSTAKA

1.1 SMP Karya Bhakti Rantau Jaya Udik II

Berdirinya SMP ini melatar belakangi bahwa pentingnya pendidikan secara mendasar masyarakat Rantau Jaya Udik II dan di Indonesia pada umumnya yang sesuai dengan program Nasional yang termaktup pada Tap MPR tentang Pendidikan Nasional. Berdirinya SMP Karya Bhakti tidak lepas dari peran Tokoh Masyarakat dan para cendikia Rantau Jaya Udik II, tepatnya pada tanggal 16 Mei 2014, Ketua Yayasan dan para dewan guru dan Bapak Kepala Desa Rantau Jaya Udik II ikut bersama mendirikan SMP Karya Bhakti dengan berstatus SMP Swasta dengan pendidrian Yayasan Karya Bhakti dengan akta Akta Notaris : YPKB No.16 Tgl.28/05/2014 yang bernaung dibawah Kantor Dinas Pendidikan kabupaten Lampung Timur ini dengan menunjuk Kepala Sekolah dan personalianya seperti gambar 2.1.

Kurikulum pengajaran yang digunakan pada SMP ini adalah kurikulum SMP tahun 2013. SMP ini memiliki ruang kelas berjumlah 5, 1 laboratorium, dan 1 perpustakaan. Belajar mengajar diajarkan oleh guru berjumlah 11 dengan siswa laki-laki berjumlah 69 orang dan siswa perempuan berjumlah 60 orang.



Gambar 2.1 Struktur Organisasi SMP Karya Bhakti Rantau Jaya Udik II

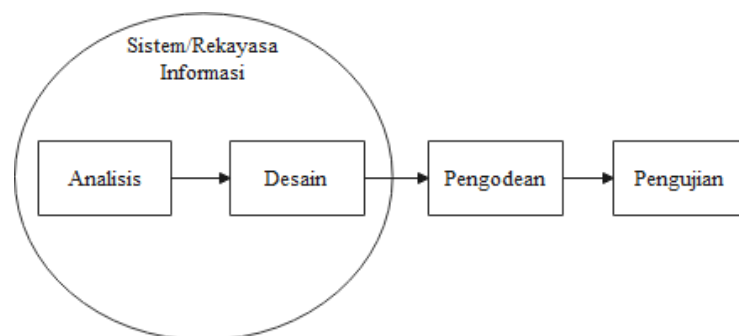
1.2 Perpustakaan Sekolah

Sekolah memiliki berbagai elemen, salah satunya adalah perpustakaan yang memiliki tanggung jawab besar dalam mengembangkan minat baca. Perpustakaan berasal dari kata dasar "pustaka" yang artinya buku. Secara umum, perpustakaan sekolah merupakan tempat yang berisi sumber informasi bahan perpustakaan yang diorganisasikan secara sistematis sehingga membantu siswa dan guru dalam Kegiatan Belajar Mengajar (KBM) di sekolah. Bahan perpustakaan yang dimaksud adalah koleksi atau bahan pustaka yang dimiliki oleh perpustakaan sekolah (Fatmawati, 2021).

Perpustakaan pada Karya Bhakti Rantau Jaya Udik II memiliki koleksi buku tidak banyak. Penelitian ini membahas buku perpustakaan yang digunakan oleh kelas 9. Jumlah buku yang tersedia di perpustakaan untuk kelas 9 sebanyak 60 buku mata pelajaran untuk siswa, 10 buku mata pelajaran untuk pegangan guru, 32 atlas, dan 24 buku pengetahuan umum.

1.3 Metode Pengembangan Sistem *Waterfall*

Model SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*) (Rosa & Shalahuddin, 2018) seperti gambar 2.2.



Gambar 2.2 Ilustrasi Model *Waterfall*

a. Analisis

Proses pengumpulan kebutuhan dilakukan secara intensif untuk mespesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

b. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

c. Pengodean

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

d. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

1.4 Desain Sistem Perangkat Lunak

Perangkat lunak adalah disiplin manajerial dan teknis yang berkaitan dengan pembuatan dan pemeliharaan produk perangkat lunak secara sistematis, termasuk pengembangan dan modifikasinya, yang dilakukan pada waktu yang tepat dan dengan mempertimbangann factor biaya (Rianto, 2021). Tujuan perancangan perangkat lunak adalah :

- a. Memperbaiki kualitas produk perangat lunak
- b. Meningkatkan produktititas
- c. Memuaskan teknisi perangkat lunak

1.4.1 MySQL

SQL (Structured Query Language) adalah bahasa pemrograman yang digunakan untuk mengelola data pada RDBMS. *SQL* dapat dipakai pada setiap sistem manajemen *database*. Untuk melakukan *query* terhadap suatu *database*, diperlukan *statemen SQL*. *Statemen* semacam ini merupakan sebuah string yang memuat kata kunci *SQL* dan informasi lain, seperti tabel *database* dan nama *file*. *Statemen* ini memberitahu mesin *database* apa informasi yang diinginkan dari *database* (Siahaan & Sianipar, 2019).

MySQL adalah RDBMS yang open source dengan dua bentuk lisensi, yaitu *free software* (perangkat lunak bebas) dan *shareware* (perangkat lunak berpemilik yang penggunaannya terbatas) Dimana setiap orang bebas menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang dijadikan closed source atau komersial. MySQL merupakan server yang melayani *database*, kita dapat mempelajari pemrograman khusus yang disebut *query* (perintah) SQL (Solichin, 2016).

1.4.2 Android Studio

Android adalah sistem operasi sumber terbuka di bawah lisensi *Apache*. Kode dengan *open source* dan lisensi-lisensi di *Android* memungkinkan perangkat lunak untuk menjadi dimodifikasi dan didistribusikan secara bebas oleh pembuat perangkat, operator nirkabel, dan aplikasi pengembang. Selain itu, *Android* memiliki sejumlah besar komunitas pengembang aplikasi (*apps*) yang memperluas fungsionalitas perangkat, umumnya ditulis dalam versi yang disesuaikan dari bahasa pemrograman *Java* (Agarina & Karim, 2019). *Android* menyertakan *kit development* perangkat lunak untuk penulisan kode asli dan perakitan modul perangkat lunak untuk membuat aplikasi bagi pengguna *Android* (Karman et al., 2019).

Android Studio merupakan Lingkungan Pengembangan Perangkat Lunak Terpadu *Integrated Development Environment (IDE)* untuk pengembangan aplikasi Android, berdasarkan IntelliJ IDEA. Selain merupakan editor kode IntelliJ dan alat pengembang yang berdaya guna, Android Studio juga menawarkan banyak fitur

untuk meningkatkan produktivitas Anda saat membuat aplikasi Android, misalnya (Herlinah & Musliadi, 2019) :

- a. Sistem versi berbasis *Gradle* yang fleksibel
- b. Emulator yang cepat dan kaya fitur
- c. Lingkungan yang menyatu untuk pengembangan bagi semua perangkat Android
- d. Instant Run untuk mendorong perubahan ke aplikasi yang berjalan tanpa membuat APK baru
- e. Kode Templat dan Integrasi GitHub untuk membuat fitur aplikasi yang sama dan mengimpor kode contoh
- f. Memiliki alat pengujian dan kerangka kerja yang ekstensif

Kotlin adalah sebuah bahasa pemrograman dengan *statically typed* (tipe statis) yang berjalan pada *Java Virtual Machine* ataupun menggunakan *Compiler LLVM* yang dapat dikompilasikan ke dalam bentuk *Source Code JavaScript*. Meskipun sintaksisnya tidak kompatibel dengan bahasa *Java*, *Kotlin* didesain untuk dapat bekerja sama dengan kode bahasa *Java* dan bergantung kepada kode bahasa *Java* dari *Java Library Class* yang ada, seperti berbagai *framework Java* yang ada. Setelah Google mengumumkan bahwa *Kotlin* menjadi bahasa kelas satu bagi Android, maka bersama *Java* dan *C++*, *Kotlin* menjadi bahasa resmi untuk pengembangan aplikasi-aplikasi Android. Kelebihan *Kotlin* dari *Java* adalah (Aljundi & Akbar, 2018):

- a. Ringkas

Kotlin secara drastis mengurangi jumlah kode *boilerplate*. Itu membuat *code* terlihat jauh lebih sederhana.

- b. Aman

Kotlin membuat seluruh *class* yang dibuat terhindar sepenuhnya dari *Null Pointer Exceptions Errors*.

- c. *Interoperable*

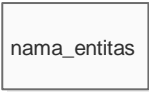
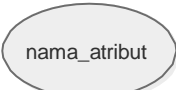
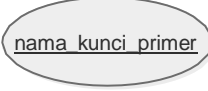
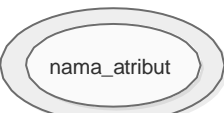


Compatible dengan *library* yang ada untuk *JVM*, *Android*, dan *browser*.

1.5 RDBMS (*Relational Database Management System*)

1.5.1 ERD (*Entity Relationship Diagram*)

ERD digunakan untuk pemodelan basis data relasional. ERD memiliki beberapa aliran notasi. Notasi yang paling banyak digunakan adalah notasi Chen (notasi yang dikembangkan oleh Peter Chen). Simbol-simbol yang digunakan pada ERD notasi Chen adalah seperti pada tabel 2.1 (Rosa & Shalahuddin, 2018).





Tabel 2.1 Simbol ERD Notasi Chen

Simbol	Keterangan
Entitas/ <i>entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer
Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam satu entitas
Atribut kunci primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan
Atribut multivalai/ <i>multivalue</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu
Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja
Asosiasi/ <i>association</i> 	Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian seperti 1 ke N atau disebut dengan <i>one to many</i>

1.5.2 DFD (*Data Flow Diagram*)

DFD adalah suatu model logika data atau proses yang dibuat untuk menggambarkan darimana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan proses tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut. Notasi yang digunakan dalam DFD (*Data Flow Diagram*) dapat di lihat pada tabel 2.2 (Rosa & Shalahuddin, 2018).




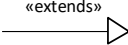
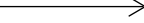

Tabel 2.2 Simbol DFD

Simbol	Keterangan
Entitas luar (<i>external entity</i>) 	Merupakan sumber atau tujuan dari aliran data dari atau ke sistem
Arus data (<i>data flow</i>) 	Mengambarkan aliran data dari satu proses ke proses lainnya
Proses (<i>process</i>) 	Proses atau fungsi yang menstransformasikan data
Simpanan data (<i>data store</i>) 	Komponen yang berfungsi untuk menyimpan data atau <i>file</i>

1.5.3 Use Case Diagram

Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Penamaan pada *use case* simpel dan mudah dipahami (Rosa & Shalahuddin, 2018). Simbol dari *use case diagram* terlihat pada tabel 2.3.


Tabel 2.3 Simbol *Use Case Diagram*

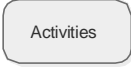


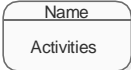

Keterangan	Simbol	Deskripsi
<i>Use Case</i>		Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja diawal-awal frase nama <i>use case</i>
Aktor		Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar itu sendiri.
Asosiasi		Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
Ekstensi		Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> , dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan.
Generalisasi		Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
<i>Include</i>		<i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan

1.5.4 Activity Diagram

Activity diagram menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis. Simbol yang terdapat pada *activity diagram* adalah seperti pada tabel 2.4 (Rosa & Shalahuddin, 2018).

Tabel 2.4 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.

Simbol	Deskripsi
	
Percabangan 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
Penggabungan 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
<i>Swimlane</i> 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

1.6 Penelitian Terdahulu

Penelitian terdahulu yang digunakan dalam menunjang penelitian yang sedang dilakukan saat ini adalah sebagai berikut :

- a. Pada penelitian berjudul “**Aplikasi Perpustakaan STMIK AUB Surakarta Berbasis Android**” dapat disimpulkan Aplikasi Perpustakaan STMIK AUB Surakarta berbasis Android, memberikan media baru bagi anggota dalam menyediakan layanan informasi koleksi buku dan peminjaman buku secara online. Selain itu juga dilengkapi dengan daftar peminjaman koleksi buku, riwayat peminjaman koleksi buku, usulan penambahan koleksi buku, berita terbaru seputar perpustakaan dan adanya notifikasi saat buku telah dikembalikan. Hasil dari penelitian ini adalah Aplikasi Perpustakaan STMIK AUB Surakarta Berbasis Android, dengan adanya aplikasi ini diharapkan dapat meningkatkan kinerja layanan informasi, transaksi peminjaman di Perpustakaan STMIK AUB (Herawati et al., 2020).
- b. Pada penelitian berjudul “**Sistem Informasi Perpustakaan Menggunakan Metode Waterfall**” menghasilkan sebuah sistem informasi perpustakaan berbasis *website* yang diujikan menggunakan *black box testing* sehingga dengan adanya sistem informasi ini akan mempermudah petugas dan anggota perpustakaan dalam

melakukan kegiatan peminjaman buku di perpustakaan serta anggota perpustakaanpun dapat mengetahui buku apa saja yang tersedia di perpustakaan dan membaca serta mendownload koleksi digital (Mailasari & Sikumbang, 2019).

- c. Pada penelitian berjudul **“Aplikasi Perpustakaan Berbasis OPAC (*Online Public Access Catalog*) di SMK N 1 Talangpadang”** menghasilkan aplikasi administrasi perpustakaan yang dapat membantu siswa dalam mencari informasi bahan pustaka dan dapat membantu pustakawan dalam mengelola administrasi perpustakaan (Prayoga et al., 2020).
- d. Pada penelitian berjudul **“Rancang Bangun Aplikasi *Dilib-Us* (*Digital Library*) Berbasis Android Menggunakan Metode *Prototyping*”** berhasil merancang aplikasi *Dilib-Us* yang dapat mengolah data buku, data anggota, data peminjaman dan pengembalian. Kecepatan pemrosesan oleh aplikasi meningkatkan pelayanan mutu padaperpustakaan serta dapat mengurangi biaya pembuatan dokumen yang tercetak dan tempat penyimpanan buku yang terbatas. Dengan aplikasi ini, member juga dimudahkan untuk mencari buku-buku yang diinginkan dengan meminjam dan mengunduh versi *online* tanpa harus datang ke perpustakaan sehingga hal ini dapat menghemat tenaga, waktu dan biaya. Penggunaan model dan data memungkinkan aplikasi dapat diterapkan di berbagai jenis perpustakaan dengan kategori buku yang berbeda-beda (Kristania & Mulyanto, 2020).
- e. Pada penelitian berjudul **“Sistem Informasi Perpustakaan di Universitas Batam Berbasis Android”** mempermudah pengguna untuk dapat melakukan peminjaman buku di perpustakaan Universitas Batam tanpa harus datang ke perpustakaan untuk memastikan buku yang akan dipinjam masih tersedia (Rahmad & Hidayat, 2020).
- f. Pada penelitian berjudul **“Penerapan Digital Library Untuk Otomatisasi Administrasi Perpustakaan”** merupakan solusi yang utama dalam menyelesaikan permasalahan pada mitra, sehingga dapat melakukan pengelolaan administrasi perpustakaan dengan memanfaatkan teknologi informasi yang ada (Megawaty et al., 2021).
- g. Pada penelitian berjudul **“Rancang Bangun Aplikasi E-Sertifikat Bina Baca Quran Mahasiswa IIB Darmajaya Berbasis Android”** dapat mengurangi

kesulitan dan kesalahan dalam mendapatkan sertifikat bina baca quran mahasiswa IIB Darmajaya. Aplikasi ini dibangun menggunakan metode pengembangan sistem waterfall dengan langkah studi literatur, pengumpulan data, menganalisis dan merancang sistem (Rosandy et al., 2022).

- h. Pada penelitian yang berjudul **“Implementasi Metode *Sequential Searching* Pada Aplikasi Rumah Kucing Berbasis *Mobile*”** mempermudah dalam mencari kucing yang lepas adopsi serta jasa pacak kucing. Aplikasi ini menerapkan algoritma *sequential searching* sebagai proses pencarian data kucing berdasarkan data lokasi sebagai kata kunci pada pencarian. Metode pengembangan sistem yang digunakan berupa prototipe dengan *tools* UML (Kurniawan & Putra, 2021).
- i. Pada penelitian yang berjudul **“Sistem *E-Library* Berbasis *Web Mobile* Pada Dinas Perpustakaan dan Kearsipan (Studi Kasus : Dinas Perpustakaan dan Kearsipan Provinsi Lampung)”** merupakan salah satu solusi untuk mengatasi kendala dari konsep perpustakaan tradisional, sehingga mampu diakses oleh pengguna tanpa batasan tempat dan waktu. Adanya teknologi informasi dapat meningkatkan minat baca dan memperluas pengetahuan tentang masyarakat. Sistem *e-library* berbasis *web* seluler dari perpustakaan daerah dan layanan arsip Lampung dirancang dengan menggunakan metode model *waterfall*. Sistem ini dibangun dengan bahasa pemrograman PHP dan MySQL sebagai basis datanya (Wulansari & Karim, 2021).
- j. Pada penelitian dengan judul **“*Tracking* Sertifikasi Produk Halal Menggunakan Algoritma *Sequential Search* Berbasis *Mobile* (Studi Kasus : LPPOM MUI Provinsi Lampung)”** menyimpulkan bahwa Sertifikat Halal adalah fatwa tertulis yang dikeluarkan oleh Majelis Ulama Indonesia (MUI) tentang status kehalalan suatu produk dengan hukum Islam. Pengusaha dapat mengajukan sertifikasi Halal pada produknya melalui LPPOM MUI. Meskipun dapat mendaftar secara *online*, tetapi pelamar tidak dapat mengetahui sejauh mana perkembangan proses sertifikasi produk bekerja. Proses sertifikasi produk dilakukan melalui telepon dengan petugas atau melalui kunjungan langsung ke Kantor LPPOM MUI Provinsi Lampung. Tentu saja, hal ini kurang praktis jika terus diterapkan, sekaligus memakannya waktu. Algoritma yang digunakan

dalam penelitian ini adalah *sequential search*. Studi ini akan menghasilkan sebuah aplikasi yang akan memungkinkan pendaftar untuk melacak kepatuhan Sertifikasi Produk Halal untuk informasi terkini mengenai proses sertifikasi produk yang didaftarkan. Berdasarkan hasil pengujian *black box* menunjukkan bahwa aplikasi yang dikembangkan dapat melakukan hal tersebut serta berjalan dengan baik dan data yang ditampilkan sesuai dengan pengolahan data yang dilakukan oleh petugas LPPOM MUI Provinsi Lampung (Ma'ruf et al., 2022).

1.7 Algoritma *Sequential Search*

Sequential search (pencarian beruntun) adalah metode yang paling mudah. *sequential search* adalah proses membandingkan setiap elemen *array* satu persatu secara beruntun yang dimulai dari elemen pertama hingga elemen yang dicari ditemukan atau hingga elemen terakhir dari *array* (R. F. Siahaan, 2020). *Sequential search* dapat dilakukan terhadap elemen *array* yang belum terurut atau terhadap elemen *array* yang terurut. Perbedaan dari keduanya terletak pada efisiensi operasi perbandingan yang dilakukan.

Dengan kata lain, *sequential search* akan mencari data dengan cara membandingkannya satu persatu dengan data yang ada. Prosesnya tentu saja akan singkat jika data yang diolah sedikit, dan akan lama jika data yang diolah banyak. Metode ini disarankan untuk digunakan pada data yang sedikit saja (Sitorus, 2015).

Contoh :

Diberikan suatu array nilai dengan banyak elemen 8 seperti berikut :

10	15	9	3	25	65	15	30	data
1	2	3	4	5	6	7	8	

Misalkan nilai yang dicari adalah $x=15$

Kalau yang diharapkan hanya menyatakan ada atau tidak ada, maka pemeriksaan hanya dilakukan terhadap 10 dan 15, karena data ditemukan atau yang dicari sama dengan 15, maka proses pencariannya berhenti sampai diangka 15 dan akan tampil "15".

1.8 Black Box Testing

Black box testing merupakan pengujian untuk mengetahui fungsi perangkat lunak yang telah berjalan sesuai dengan kebutuhannya. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan pengeluaran perangkat lunak sesuai dengan spesifikasi yang dibutuhkan (Rosa & Shalahuddin, 2018). *Black box testing* mencoba untuk menemukan kesalahan dalam kategori berikut :

- a. Fungsi tidak benar atau hilang.
- b. Kesalahan *interface* atau antarmuka.
- c. Kesalahan dalam struktur data atau akses *database* eksternal