

BAB II

LANDASAN TEORI

2.1 Pengertian Aplikasi

2.1.1 Aplikasi

Aplikasi adalah suatu subkelas dari suatu perangkat lunak komputer yang memanfaatkan kemampuan komputer secara langsung untuk melakukan suatu tugas yang diinginkan pengguna. Aplikasi dapat juga dikatakan sebagai penerjemah perintah-perintah yang dijalankan pengguna komputer untuk diteruskan ke atau diproses oleh perangkat keras. Selain bisa membantu dan mempercepat proses pekerjaan manusia, aplikasi juga bisa menciptakan hasil yang lebih akurat dalam memecahkan permasalahan. Aplikasi pada umumnya digunakan untuk mengontrol perangkat keras (yang sering disebut sebagai device driver), melakukan proses perhitungan, dan berinteraksi dengan aplikasi yang lebih mendasar lainnya (seperti sistem operasi, dan bahasa pemrograman)[3].

2.1.2 Buta Warna

Buta warna merupakan suatu keadaan dimana seseorang tidak dapat membedakan warna tertentu yang biasanya bisa dibedakan oleh orang lain dengan mata yang normal. Buta warna dibagi menjadi tiga: *monokromatis* (buta warna total), *dikromasi* (hanya dua sel kerucut yang berfungsi), dan *anomalous trikromasi* (tiga sel kerucut berfungsi, salah satunya kurang baik). Dari semua jenis buta warna, kasus yang paling umum ditemukan adalah anomalous trikromasi (buta warna parsial), khususnya *deutranomali* atau kurang berfungsinya penglihatan warna terhadap warna merah maupun hijau[4].

Gangguan persepsi warna atau buta warna dibagi kedalam beberapa jenis yaitu bawaan dan didapat maupun faktor lain seperti karena *Shaken Baby Syndrome*, cedera atau trauma pada otak dan retina, maupun pengaruh sinar ultra violet. Buta warna yang diturunkan secara genetik dibawa oleh kromosom X pada perempuan, dan diturunkan pada anak-anaknya. Kelainan buta warna kebanyakan menyerang laki-laki karena sifatnya yang genetik terkait pada kromosom X. Ketika seseorang mengalami buta warna, mata mereka tidak mampu menghasilkan keseluruhan pigmen yang dibutuhkan untuk mata berfungsi dengan normal.

Ada beberapa penyebab seseorang mengalami buta warna, diantaranya adalah:

1. Penyakit : Terdapat sejumlah penyakit yang bisa menyebabkan buta warna, seperti penyakit parkinson, penyakit *alzheimer*, *glaukoma*, *neuritis optik*, *leukemia*, *diabetes*, pecandu alkohol kronis, *macular degeneration*, dan anemia sel sabit[5].
2. Usia : Kemampuan seseorang untuk membedakan warna perlahan-lahan akan berkurang seiring bertambahnya usia. Ini adalah hal yang alami dalam proses penuaan dan tidak perlu dicemaskan secara berlebihan.
3. Faktor genetika : Kebanyakan penderita buta warna mengalaminya sejak lahir dan merupakan faktor genetika yang diturunkan oleh orang tua. Penderita buta warna akibat faktor genetika jauh lebih sering terjadi pada pria dibandingkan wanita.
4. Bahan kimia : Seseorang bisa mengalami buta warna jika terpapar bahan kimia beracun misalnya di tempat kerja, seperti karbon *disulfida* dan pupuk.
5. Efek samping pengobatan tertentu : Beberapa pengobatan berpotensi menyebabkan buta warna, seperti *digoxin*, *pheytoin*, *chloroquine* dan *sildenafil* yang juga dikenal sebagai *Viagra*. Jika gangguan disebabkan oleh pengobatan, biasanya pandangan akan kembali normal setelah berhenti mengkonsumsi obat.

2.1.3 Klasifikasi Buta Warna

Tingkatan buta warna dapat diklasifikasikan menjadi 3 jenis yang sering ditemui dan diketahui yaitu :

1. *AnomaliTrikoma* suatu keadaan dimana tiga jenis sel kerucut tetap ada, tetapi satu diantaranya tidak normal atau tidak berfungsi dengan baik, sehingga penderita akan mengalami kesulitan membedakan nuansa warna tertentu. Jenis buta warna ini sering dialami oleh kebanyakan orang yaitu:

- a. *Protanomali* (Lemah Merah)

Terjadi karena sel kerucut warna merah tidak berfungsi dengan baik, sehingga penderita kurang sensitif atau kesulitan mengenali warna merah dan perpaduannya.

b. *Deuteranomali* (Lemah Hijau)

Terjadi karena sel kerucut warna hijau tidak berfungsi dengan baik, sehingga penderita kurang sensitif atau kesulitan mengenali warna merah dan perpaduannya.

c. *Tritanomali* (Lemah Biru)

Terjadi karena sel kerucut warna biru tidak berfungsi dengan baik, sehingga penderita kurang sensitif atau kesulitan mengenali warna merah dan perpaduannya.

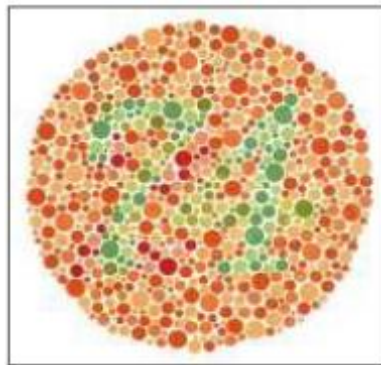
2. *Dikhromat* keadaan ketika satu dari tiga sel kerucut tidak ada. Ada tiga klasifikasi dikromasi yaitu:

1. *Protanopia* (buta warna merah) Protanopia terjadi karena sel kerucut warnamerah tidak ada sehingga tingkat kecerahan warna merah atau perpaduannya menjadi berkurang.
2. *Deuteranopia* (buta warna hijau) terjadi karena sel kerucut warna hijau tidak ada sehingga tingkat kecerahan warna hijau atau perpaduannya menjadi berkurang.
3. *Tritanopia* (buta warna biru) Tritanopia terjadi karena sel kerucut warna biru tidak ada sehingga tingkat kecerahan warna biru atau perpaduannya menjadi berkurang.
4. *Monokhromat* kondisi retina mata yang mengalami kerusakan total dalam merespon warna. *Monokromat* ditandai dengan hilangnya atau berkurangnya semua penglihatan warna, sehingga yang terlihat hanya putih dan hitam yang mampu diterima retina. Jenis buta warna ini prevalensinya sangat jarang.

2.1.4 Metode Ishihara

Metode Ishihara ini di kembangkan menjadi Tes Buta Warna Ishihara oleh Dr. Shinobu Ishihara. Tes ini pertama kali dipublikasi pada tahun 1917 di Jepang

dan terus digunakan di seluruh dunia, sampai sekarang. Tes buta warna Ishihara terdiri dari lembaran yang didalamnya terdapat titik-titik dengan berbagai warna dan ukuran. Titik berwarna tersebut disusun sehingga membentuk lingkaran. Warna titik itu dibuat sedemikian rupa sehingga orang buta warna tidak akan melihat perbedaan warna seperti yang dilihat orang normal (pseudo-isochromatism)[6].



Gambar 2.1 *Plate* Metode Ishihara

2.1.5 Tahapan Dalam Pemeriksaan Tes Buta Warna

Tahapan dalam pemeriksaan buta warna dengan metode ishihara sebagai berikut :



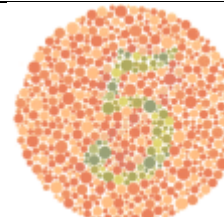
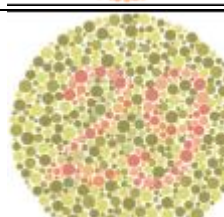
1. Menggunakan buku Ishihara, Ruang pemeriksaan harus cukup pencahayaannya dan lama pengamatan untuk membaca angka masing-masing lembar maksimum 10 detik.

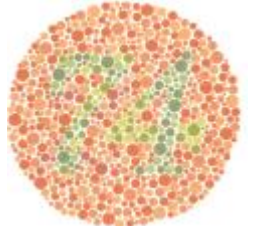
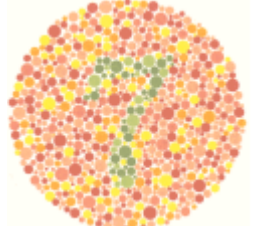
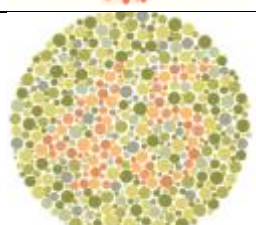
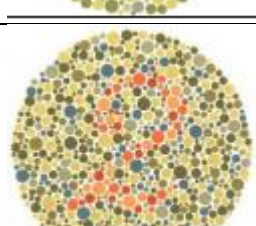
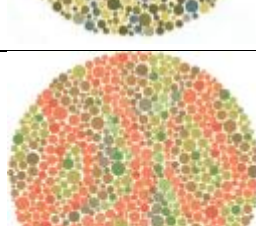
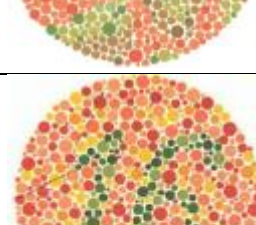
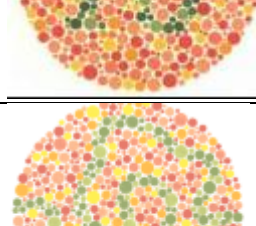
Pada tes pembacaan buku Ishihara dapat disimpulkan Normal atau tidak buta warna, Buta warna total pada *plate* no. 28 dan 29, untuk orang normal tidak bisa menunjukkan adanya alur, sedangkan untuk penderita buta warna parsial dapat menunjukkan adanya alur dari satu sisi ke sisi yang lainnya. Buta Warna Parsial sebagai berikut :




- a. Bila *plate* no. 1 sampai dengan no 17. hanya terbaca 13 *plate* atau kurang.
- b. Bila terbaca angka-angka pada *plate* no. 18, 19, 20 dan 21 lebih mudah atau lebih jelas dibandingkan dengan *plate* no. 14, 10, 13, dan 17.
- c. Bila ragu-ragu kemungkinan buta warna parsial dapat dites dengan cara sebagai berikut :

- Membaca angka-angka pada plate no. 22, 23, 24, dan 25. Pada orang normal, akan terbaca dengan benar angka-angka pada plate-plate tersebut diatas secara lengkap (dua rangkap). Pada penderita buta warna parsial hanya terbaca satu angka pada tiap-tiap plate tersebut diatas.
- Menunjuk arah alur pada plate no. 26, 27, 30, 31, 32, 33, 34, 35, 36, 37, dan 38. Untuk orang normal bisa menunjuk alur secara benar sedangkan untuk buta warna parsial dapat menunjukkan adanya alur dari satu sisi yang lainnya.

Tabel 2.1 Tes Buta Warna dengan 14 *Plate* Ishihara

NO. Plate	Gambar Plate	Mata Normal	Mata Buta Warna Parsial	Mata Buta Warna Total
1		Melihat angka "12"	Melihat angka "12"	Melihat angka "12"
2		Melihat angka "8"	Melihat angka "3"	Tidak ada
3		Melihat angka "5"	Melihat angka "2"	Tidak ada
4		Melihat angka "29"	Melihat angka "70"	Tidak ada

5		Melihat angka "74"	Melihat angka "21"	Tidak ada
6		Melihat angka "7"	Tidak ada	Tidak ada
7		Melihat angka "45"	Tidak ada	Tidak ada
8		Melihat angka "2"	Tidak ada	Tidak ada
9		Tidak ada	Melihat angka "2"	Tidak ada
10		Melihat angka "16"	Tidak ada	Tidak ada
11		Melihat jalur hijau	Tidak ada	Tidak ada

12		Melihat angka "35"	Mata normal dan mereka dengan defisiensi merahhijau ringan melihat gambar "35" tetapi protanopia dan protanomalia yang kuat akan membaca "5" saja dan deuteranopia dan deuteranomalia kuat membaca "3" saja.	
13		Melihat angka "96"	Mata normal dan mereka dengan defisiensi merahhijau ringan melihat gambar "96" tetapi protanopia dan protanomalia yang kuat akan membaca "6" saja dan deuteranopia dan deuteranomalia yang kuat hanya membaca "9"	
14		Melihat garis ungu dan merah	Untuk protanopia dan protanomalia hanya melihat garis ungu. Deuteranopia dan deuteranomalia hanya melihat garis merah	Tidak ada

2.1.6 Database

Beberapa defenisi tentang *database* dari beberapa orang ahli, *database* adalah sebagai berikut :

1. *Database* adalah sekumpulan data *store* yang tersimpan dalam magnetic disk, optical disk, magnetic drum atau media penyimpanan sekunder lainnya[7].

2. *Database* adalah sekumpulan program-program aplikasi umum yang mengeksekusi dan memproses data secara umum seperti pencarian data, peremajaan data, penambahan dan penghapusan data. *Database* terdiri dari data yang akan digunakan atau diperuntukkan terhadap *user*, dimana masing-masing *user* akan menggunakan data tersebut sesuai dengan tugas dan fungsinya, dan *user* lain juga dapat menggunakan data tersebut dalam waktu yang bersama.

2.1.7 Web

WEB atau juga dikenal dengan *World Wide Web* atau *WWW* adalah salah satu layanan yang didapat oleh pemakai komputer yang terhubung ke internet. *Web* ini menyediakan informasi bagi pemakai komputer yang terhubung ke internet dari sekedar informasi “sampah” atau informasi yang tidak berguna sama sekali sampai informasi yang serius, dari informasi yang gratisan sampai informasi yang komersial. *Website* atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi *teks*, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya itu baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*)[8].

2.2 Metode Perancangan Sistem

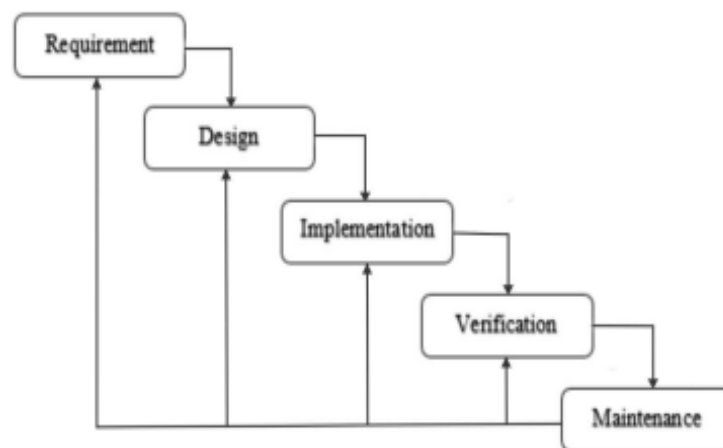
2.2.1 Metode Waterfall

Metode air terjun atau yang sering disebut metode waterfall sering dinamakan siklus hidup klasik (*classic life cycle*), nama model ini sebenarnya adalah “*Linear Sequential Model*” dimana hal ini menggambarkan pendekatan yang sistematis dan juga berurutan pada pengembangan perangkat lunak, dimulai dengan spesifikasi kebutuhan pengguna lalu berlanjut melalui tahapan-tahapan perencanaan (*planning*), permodelan (*modelling*), konstruksi (*construction*), serta penyerahan sistem ke para pengguna (*deployment*), yang diakhiri dengan dukungan pada perangkat lunak lengkap yang dihasilkan[9].

Model *waterfall* pertama kali diperkenalkan oleh Winston Royce sekitar tahun 1970 sehingga sering dianggap kuno, tetapi merupakan model yang paling banyak dipakai didalam *Software Engineering* (SE). saat ini model *waterfall* merupakan model pengembangan perangkat lunak yang sering digunakan. Model pengembangan ini melakukan pendekatan secara sistematis dan berurutan. Disebut *waterfall* karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan. Model pengembangan ini bersifat linear dari tahap awal pengembangan sistem yaitu tahap perencanaan sampai tahap akhir pengembangan sistem yaitu tahap pemeliharaan. Tahapan berikutnya tidak akan dilaksanakan sebelum tahapan sebelumnya selesai dilaksanakan dan tidak bisa kembali atau mengulang ke tahap sebelumnya.

2.2.2 Tahapan Metode *Waterfall*

Untuk tahapan dan penjelasan dari metode *Waterfall* dapat dilihat pada gambar dibawah ini:



Gambar 2.2 Metode *Waterfall*

1. *Requirement*

Tahap ini pengembang sistem diperlukan komunikasi yang bertujuan untuk memahami perangkat lunak yang diharapkan oleh pengguna dan batasan perangkat lunak tersebut. Informasi dapat diperoleh melalui wawancara, diskusi atau survei langsung. Informasi dianalisis untuk mendapatkan data yang dibutuhkan oleh pengguna.

2. *Design*

Pada tahap ini, pengembang membuat desain sistem yang dapat membantu menentukan perangkat keras (*hardware*) dan sistem persyaratan dan juga membantu dalam mendefinisikan arsitektur sistem secara keseluruhan.

3. Implementation

Pada tahap ini, sistem pertama kali dikembangkan di program kecil yang disebut unit, yang terintegrasi dalam tahap selanjutnya. Setiap unit dikembangkan dan diuji untuk fungsionalitas yang disebut sebagai unit testing.

a. Verification

Pada tahap ini, sistem dilakukan verifikasi dan pengujian apakah sistem sepenuhnya atau sebagian memenuhi persyaratan sistem, pengujian dapat dikategorikan ke dalam unit testing (dilakukan pada modul tertentu kode), sistem pengujian (untuk melihat bagaimana sistem bereaksi ketika semua modul yang terintegrasi) dan penerimaan pengujian (dilakukan dengan atau nama pelanggan untuk melihat apakah semua kebutuhan pelanggan puas).

b. Maintenance

Pada tahap ini adalah tahap akhir dari metode *waterfall*. Perangkat lunak yang sudah jadi dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya.

2.2.3 Kelebihan Metode Waterfall

Adapun beberapa kelebihan yang didapat dari metode *waterfall* adalah sebagai berikut :

1. Kualitas dari sistem yang dihasilkan akan baik, karena pelaksanaannya dilakukan secara bertahap.
2. Proses pengembangan model *fase one by one*, sehingga meminimalis kesalahan yang mungkin akan terjadi.

2.2.4 Kekurangan Metode Waterfall

Ada beberapa kekurangan yang terdapat di metode *waterfall* di antaranya adalah sebagai berikut :

1. Waktu pengembangan lama dan biayanya mahal.
2. Diperlukan manajemen yang baik, karena proses pengembangan tidak dapat dilakukan secara berulang sebelum terjadinya suatu produk.
3. Kesalahan kecil akan menjadi masalah besar jika tidak diketahui sejak awal pengembangan yang berakibat pada tahapan selanjutnya.
4. Pada kenyataannya, jarang mengikuti urutan sekuensial (runtutan) seperti pada teori. *Iterasi* (perulangan) sering terjadi menyebabkan masalah baru.

2.3 Alat Bantu Perancangan

Dalam perancangan sistem membutuhkan beberapa alat bantu perancangan agar analisa dan hasil yang ingin dicapai dapat mencapai sebuah hasil maksimal berikut alat bantu yang digunakan :


2.3.1 *Unified Modelling Language*



Unified Modelling Language yaitu “Bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung”. UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak[10].

2.3.2 *Use Case Diagram*

Use Case Diagram merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat”. Setiap *use case* dilengkapi dengan skenario. Skenario *use case* adalah alur jalannya proses *use case* dari sisi aktor dan sistem.

Tabel 2.2 *Simbol Use Case Diagram*

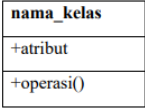



Simbol	Keterangan
 <p><i>Actor</i></p>	<ul style="list-style-type: none"> • Orang atau sistem yng berinteraksi dengan sistem informasi usulan. • Aktor ditempatkan diluar batas system.




 Use Case	<ul style="list-style-type: none"> • Mewakili sebagian besar sistem secara fungsional. • Ditempatkan di dalam batas sistem.
 System	<ul style="list-style-type: none"> • Menspesifikasikan paket yang menampilkan sistem secara terbatas.

2.3.3 Class Diagram

Class Diagram merupakan inti dari proses pemodelan objek. *Tool* yang membantu dalam visualisasi struktur kelas dari suatu sistem. Hal ini disebabkan karena class adalah deskripsi kelompok objek-objek dengan *property*, perilaku (operasi) dan relasi yang sama. Disamping itu *class diagram* bisa memberikan pandangan global atas sebuah sistem.

Tabel 2.3 Simbol *Class Diagram*


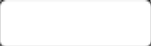
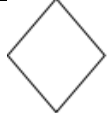

Simbol	Keterangan
 Kelas	Kelas pada struktur sistem.
 Antarmuka/Interface	Sama dengan konsep interface dalam pemrograman berorientasi objek
 Asosiasi / Association	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity.
 Asosiasi Berarah	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan multiplicity.



 Generalisasi	Relasi antar kelas dengan makna generalisasi spesialisasi (umum khusus).
 Kebergantungan	Kebergantungan antar kelas.
 Agregasi / Aggregation	Relasi antar kelas dengan makna semua bagian (<i>whole part</i>).

2.3.4 Activity Diagram

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada di perangkat lunak.

Tabel 2.4 Simbol *Activity Diagram*

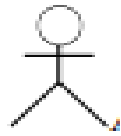

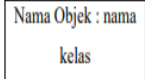
Simbol	Keterangan
 Status awal	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
 Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
 Percabangan / Decision	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
 Status akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.






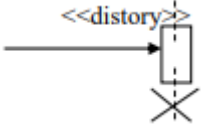
 Pengabungan / Join	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
 Swimlane	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

2.3.5 Sequence Diagram

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek.

Tabel 2.5 Simbol *Sequence Diagram*

Simbol	Keterangan
 Aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat, walaupun simbol aktor itu orang belum tentu merupakan orang biasanya dinyatakan dengan menggunakan kata kerja frase nama <i>use case</i> .
 Garis Hidup	Menyatakan kehidupan suatu objek.
 Objek	Menyatakan objek yang berinteraksi pesan.

 <p style="text-align: center;">Waktu Aktif</p>	<p>Menyatakan Objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif.</p>
 <p style="text-align: center;">Pesan Tipe <i>Create</i> <<Create>></p>	<p>Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.</p>
 <p style="text-align: center;">Pesan Tipe <i>Call</i> 1: Nama_Metode()</p>	<p>Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lain atau diri sendiri.</p>
 <p style="text-align: center;">Pesan Tipe <i>Send</i> 1 : Masukan</p>	<p>Menyatakan bahwa suatu objek mengirimkan data / masukkan / informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.</p>
 <p style="text-align: center;">Pesan Tipe <i>Return</i> 1: Keluaran</p>	<p>Menyatakan bahwa suatu objek telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.</p>
 <p style="text-align: center;">Pesan Tipe <i>Destroy</i></p>	<p>Menyatakan bahwa suatu objek mengakhiri hidup objek lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i>.</p>

2.4 Alat Bantu Pemrograman

2.4.1 Hypertext Preprocessor (PHP)

Hypertext Preprocessor yang digunakan sebagai *script sever-side* dalam pengembangan *web* yang disisipkan pada dokumen HTML, PHP dapat juga

diartikan sebagai sebuah bahasa scripting yang terpasang pada HTML dimana sebagian sintaksnya mirip dengan bahasa C, Java, ASP, dan Perl ditambah beberapa fungsi PHP yang spesifik. PHP dapat diintegrasikan (*embedded*) ke dalam *web server*, atau dapat berperan sebagai program CGI yang terpisah. Karakteristik yang paling unggul dan paling kuat dari PHP adalah lapisan integrasi database (*database integration layer*). Database yang didukung PHP adalah: *Oracle, Adabas-D, Sybase, FilePro, mSQL, Velocis, MySQL, Informix, Solid, dBase, ODBC, Unix dbm* dan *PostgreSQL*[11].

PHP sendiri merupakan sebuah bahasa pemrograman yang dipakai untuk membuat *coding website* dinamis yang memungkinkan kita untuk melakukan *update* pada *website* setiap saat. Berbeda halnya dengan markah HTML yang memuat *source code* yang ditampilkan didalam *website*, sedangkan source code PHP sendiri tidak di tampilkan didalam halaman suatu website dikarenakan PHP diproses dan diolah didalam sebuah *server*. Dalam *database* PHP memiliki kedinamisan yang bisa dihubungkan langsung dengan *MySQL, Oracle*. Namun untuk pemrograman *websitenya* sendiri yang digunakan adalah *MySQL*. PHP sampai saat ini telah mengalami kemajuan yang begitu pesat dan untuk mengawali kode dalam PHP digunakan *code <? Dan di tutup dengan code ?*.

2.4.2 MySQL

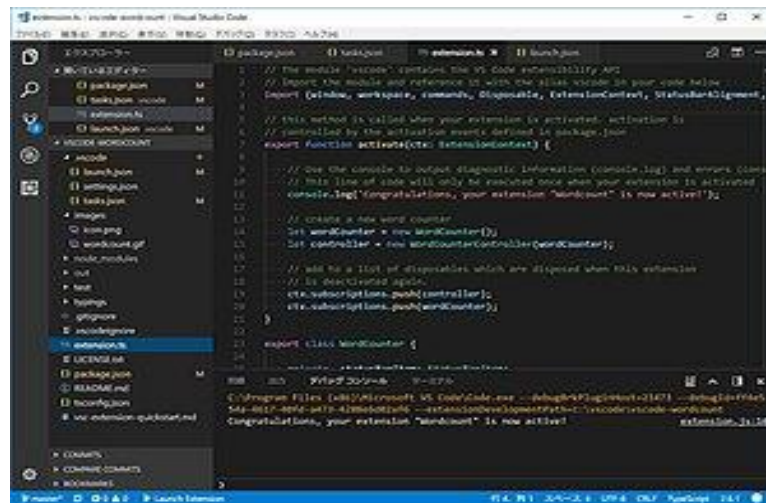
MySQL adalah perangkat lunak untuk sistem manajemen *database (database management sistem)*, *MySQL* merupakan *software database* yang paling populer dikarenakan performa *query* dari database *MySQL* dapat dikatakan paling cepat dan bisa dibbilang jarang bermasalah. *MySQL* merupakan *software* yang tergolong

sebagai DBMS (*database management system*) yang bersifat *open source* setiap pengguna dapat secara bebas menggunakannya, *MySQL* adalah sebuah konsep pengoperasian basis data, terutama untuk pemilihan atau seleksi dan pemasukan data yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

2.4.3 Visual Studio Code

Visual Studio Code (*VS Code*) adalah sebuah teks *editor* ringan dan handal yang dibuat oleh *Microsoft* untuk sistem operasi *multiplatform*, artinya tersedia juga untuk versi *Linux*, *Mac*, dan *Windows*. Teks *editor* ini secara langsung mendukung bahasa pemrograman *Javascript*, *Typescript*, dan *Nodejs*, serta bahasa pemrograman lainnya dengan bantuan *plugin* yang dapat dipasang via *marketplace Visual Code* (seperti *C++*, *C#*, *Python* dan *Java*[12]).

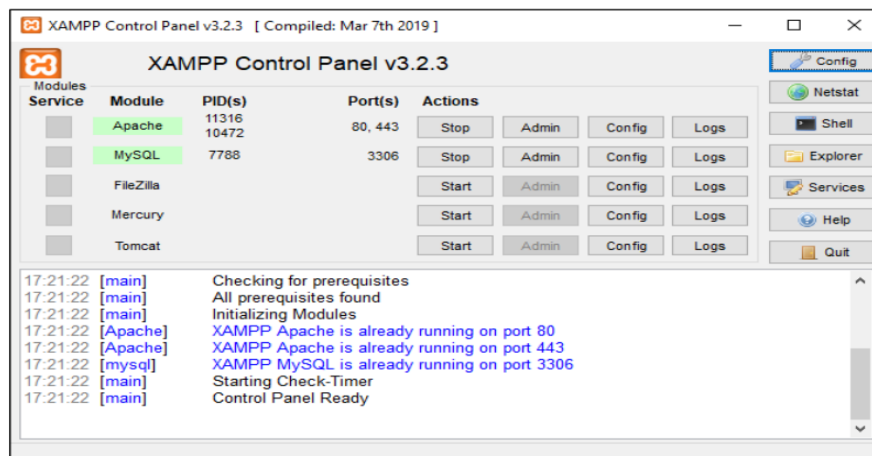
Teks *editor Visual Studio Code* juga bersifat *open source code* yang mana *code* sumbernya dapat kalian lihat dan kalian dapat berkontribusi untuk pengembangannya. Kode sumber dari *Visual Studio Code* ini dapat dilihat di *link github* . hal ini juga membuat *Visual Studio Code* menjadi favorit para pengembang aplikasi karena para pengembang aplikasi bisa ikut serta dalam proses pengembangan *Visual Studio Code* ke depannya.



Gambar 2.3 Tampilan Aplikasi Visual Studi Code

2.4.4 XAMPP

XAMPP adalah sebuah paket perangkat lunak (*software*) komputer yang sistem penamaannya diambil dari akronim kata *Apache*, *PHP*, dan *Perl*. Xampp nantinya kami gunakan sebagai aplikasi pembantu dalam menampilkan hasil source code yang kami buat. Aplikasi ini juga berfungsi untuk mengelola *database* yang akan kami gunakan.



Gambar 2.4 Tampilan Aplikasi XAMPP

2.5 Metodologi Penelitian

2.5.1 Penelitian Terdahulu

Topik penelitian ini juga pernah diangkat oleh beberapa peneliti sebelumnya, penelitian terdahulu berguna untuk mengetahui bagaimana metode penelitian dan

hasil-hasil penelitian yang dilkauan. Penelitian terdahulu digunakan sebagai tolak ukur peneliti untuk menganalisis suatu penelitian. Tujuan dari penelitian terdahulu ini adalah guna untuk mengetahui langkah penulisan salah atau benar nya dan referensi peneliti sebagai acuan dalam melakukan penelitian untuk menghasilkan penelitian secara maksimal.

Tabel 2.6 Referensi Penelitian Terdahulu

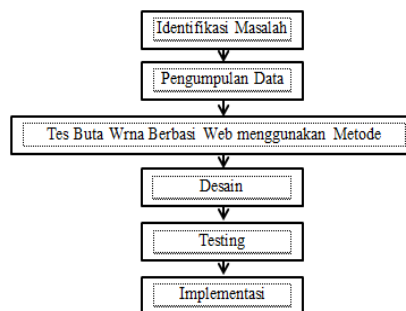
NO	Peneliti	Tujuan Penelitian	Hasil Penelitian
1	Eka Permana , Sella Tamara, Jurnal Teknologi Informasi dan Komunikasi ISSN: 2252-4517 STMIK Subang, April 2013, “Implementasi Sistem Berbasis Web Untuk Visualisasi Tes Buta Warna (Colorblind Test)”.	Menentukan buta warna yang dialami seseorang dengan menghasilkan kesimpulan yaitu normal, buta warna parsial dan buta warna total, dan hasil dari tes tersebut langsung tersimpan kedalam suatu database komputer	Dengan pembuatan aplikasi penentuan buta warna ini diharapkan dapat memperoleh pelayanan kesehatan yang cukup efisien dan efektif sesuai dengan apa yang diharapkan oleh user/pemakai. 2. Dengan sistem aplikasi ini diharapkan mampu mengenali buta warna sedini mungkin serta dapat memberikan jawaban atau keputusan yang sesuai serta

			akurat. 3. Aplikasi ini sangat berguna dan diharapkan dapat bermanfaat bagi semua kalangan yang menggunakannya.
2	Made Hanindia Prami Swari, I Gede Yudhi Mahardika, Faisal Muttaqin, Isbn : 978-6-02-526748-2 – Seminar Santika 4-5 September 2019, “Penggunaan Plat Digital Dalam Pengembangan Aplikasi Pendeteksi Buta Warna”.	Membangun sebuah aplikasi pendeteksi buta warna menggunakan metode Ishihara Test	Dengan dibangunya Aplikasi Pendeteksi Buta Warna Menggunakan Metode Ishihara Test ini dapat mempermudah merawat cetakan gambar atau buku itu sendiri, karena cetakan gambar (plate) sudah di rubah ke dalam bentuk gambar digital. Kemungkinan untuk pudar dan tercoret sangatlah kecil.
3	Mustika Ayu Nadilah, Rissya Mutya Prima, Taufiq	Mengatasi masalahan maka dibuat aplikasi yang	Aplikasi Tes Buta Warna Berbasis Web di Puskesmas Kecamatan Hajimena, dapat

	Hidayatulloh, Agiska Ria Supriyatna, Karya Ilmiah Mahasiswa Manajemen Informatika 2019, “Aplikasi Tes Buta Warna Berbasis Web Puskesmas Kecamatan Hajimena”.	akan membantu pada proses pendaftaran, pengecekan, dan pembuatan surat keterangan tidak buta warna atau rujukan, sehingga lebih efisien waktu.	membantu menangani masalah pengecekan tes buta warna yang ada di puskesmas dan memberikan informasi awal kepada pengguna setelah melakukan pengecekan tes buta warna, juga menghindari adanya kecurangan pada saat tes karena sistem penilaian tes yang dilakukan oleh sistem.
4	Rico Estrada, Eko Adi Sarwoko, “Aplikasi Tes Buta Warna Dengan Metode Ishihara dan Farnsworth Munsell D-15 (Studi Kasus : Puskesmas Rowosari)”.	Menghasilkan sebuah aplikasi tes buta warna dengan metode Ishihara dan Farnsworth Munsell D-15.	Mempermudah petugas tes buta warna dalam melakukan tes buta warna dan juga mempermudah dalam melihat data yang lalu serta menghasilkan laporan yang dapat langsung dicetak melalui printer.

2.5.2 Kerangka Kerja Penelitian

Kerangka kerja penelitian merupakan dasar yang digunakan untuk memecahkan atau menangani suatu masalah kompleks. Kerangka kerja ini merupakan langkah-langkah yang akan dilakukan dalam penyelesaian masalah yang akan dibahas, adapun kerangka kerja penelitian dapat digambarkan pada gambar berikut :



Gambar 2.5 Kerangka Kerja Penelitian

Berdasarkan **Gambar 2.5** dijabarkan urutan-urutan dalam langkah kerja yaitu sebagai berikut :

1. Identifikasi Masalah

Identifikasi masalah dalam penulisan penelitian ini adalah bagaimana merancang aplikasi tes buta warna berbasis web menggunakan metode isihara.

2. Pengumpulan Data

a. Observasi

Mengunjungi secara langsung ke Klinik Mata Dr. Sri Ninin Asnita yang belamat di Jl. Siringo-ringo No. 14 dan melihat kejadian yang terjadi di Klinik Mata yang berkaitan dengan tes buta warna.

b. Wawancara

Yaitu suatu teknik pengumpulan data dengan cara wawancara atau melakukan tanya jawab dengan pihak Klinik Mata mengenai pemeriksaan tes buta warna yang akan di peneliti sampaikan dalam penelitian. Wawancara dilakukan kepada Dr. Sri Ninin Asnita SpM selaku Dr mata di Klinik Mata tersebut.

c. Studi Pustaka

Yaitu pengumpulan data berdasarkan sumber-sumber yang biasa di gunakan untuk penulisan yang berkaitan dengan judul.

3. Tes Buta Warna Berbasis Web menggunakan Metode Ishihara

Pada tahap ini melakukan tes buta warna menggunakan metode ishihara berbasis web yang mempermudah dalam melakukan tes buta warna untuk mengetahui seseorang buta warna atau tidak.

4. Desain

Pada tahap ini melakukan pendesainan dalam perancangan aplikasi tes buta warna berbasis web menggunakan metode Isihihara dengan mengatur tata letak dan mempermudah untuk dipakai oleh penngguna baik yang sudah profesional atau pun yang masih awam.

5. Tahap *Testing*

Pada tahap ini yaitu melaukan tahap pengembangan perangkat lunak yang harus dilakukan untuk memastikan bahwa program aplikasi tes buta warna berbasis web menggunakan metode Ishihara yang telah di rancang penulis sudah sesuai dengan kebutuhan atau tidak.

6. Tahap Implementasi

Pada tahap ini merupakan tahap sekaligus pengujian bagi rancangan baru serta merupakan tahap dimana aplikasi siap dioperasikan pada keadaan yang sebenarnya, efektifitas yang baru akan dikathui secara pasti, juga untuk semua kelebihan dan kekurangan rancangan program aplikasi.