



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB II

LANDASAN TEORI

2.1 Pengertian Genetika

Terdapat beberapa pengertian genetika yang telah disampaikan oleh para ahli, diantaranya adalah sebagai berikut:

- Menurut Saktiyono (2002), genetika merupakan ilmu yang mempelajari bagaimana ciri atau sifat dari orangtua yang diturunkan atau diwariskan kepada keturunannya.
- Menurut Istamar Syamsuri (2007), genetika merupakan ilmu yang mempelajari pewarisan sifat dari induk kepada keturunannya dengan mengikuti pola – pola tertentu.
- Menurut Neil . A. Cambell dan Jane. B. Reece (2008), genetika merupakan bidang sains yang mempelajari pewarisan sifat dari induk ke keturunannya dengan variasi yang diwariskan.
- Menurut D.A. Pratiwi (2006), genetika merupakan ilmu yang mempelajari penurunan sifat – sifat genetik (hereditas).

Berdasarkan pengertian menurut para ahli, dapat disimpulkan bahwa genetika adalah ilmu sains yang mempelajari pewarisan ciri dan sifat genetik dari induk kepada keturunannya dengan pola dan variasi tertentu.

2.2 Rekayasa Genetika

Pada bagian ini akan menjelaskan tentang pengertian dari rekayasa genetika, teknik rekayasa genetika, dan produk rekayasa genetika.

2.2.1 Pengertian Rekayasa Genetika

Terdapat beberapa pengertian rekayasa genetika yang telah disampaikan oleh para ahli, diantaranya adalah sebagai berikut:

- Menurut Saktiyono (2004), rekayasa genetika merupakan suatu proses penggantian atau penambahan pada organisme DNA asli dari sel organisme lainnya.
- Menurut Istamar Syamsuri (2007), rekayasa genetika merupakan suatu proses mengubah sifat makhluk hidup melalui perubahan gen organisme tersebut.
- Menurut Neil . A. Cambell dan Jane. B. Reece (2008), rekayasa genetika merupakan suatu proses yang dilakukan pada gen organisme dengan memanipulasi langsung gen tersebut demi mencapai tujuan praktis.

Berdasarkan pengertian menurut para ahli, dapat disimpulkan bahwa rekayasa genetika adalah suatu proses mengubah sifat makhluk hidup melalui penggantian atau penambahan gen organisme untuk mencapai tujuan tertentu.

2.2.2 Teknik Rekayasa Genetika

Menurut Saktiyono (2004), teknik rekayasa genetika dapat dilakukan dengan cara pecangkakan dan pencampuran DNA. Hal tersebut tentu harus melewati berapa proses, yaitu:

- DNA organisme asli dan DNA organisme lain harus dipotong terlebih dahulu dengan menggunakan enzim restriksi.
- Setelah masing - masing DNA organisme terpotong, DNA dari kedua organisme akan dicampurkan dalam suatu wadah.
- Dalam waktu yang cukup lama DNA suatu organisme tersebut akan tersisipi dengan DNA organisme lainnya.
- Hasilnya sel – sel organisme tersebut mengandung DNA yang telah direkayasa.

2.2.3 Produk Rekayasa Genetika

Menurut Saktiyono (2004), rekayasa genetika merupakan suatu bioteknologi modern yang telah menghasilkan banyak produk yang bermanfaat untuk manusia. Dari banyaknya produk hasil rekayasa genetika, ada 5 produk utama yang telah terbukti banyak membantu tubuh manusia, yaitu:

- Vaksin Hepatitis yang berfungsi untuk mencegah terkena penyakit radang hati (hepatitis).
- Hormon Insulin yang berfungsi untuk menolong penderita kencing manis (diabetes melitus).
- Interferon yang berfungsi untuk pengobatan penyakit yang disebabkan oleh virus dan kanker.

- Antibodi Monoklonal yang berfungsi sebagai antibodi untuk mencegah berbagai macam penyakit yang masuk tubuh.
- Penggabungan Protoplasma yang berfungsi untuk mengatasi penyakit tanaman dalam bidang pertanian.

2.3 Reptil

Pada bagian ini akan menjelaskan tentang pengertian dari reptil, dan ordo reptil.

2.3.1 Pengertian Reptil

Terdapat beberapa pengertian reptil yang telah disampaikan oleh para ahli, diantaranya adalah sebagai berikut:

- Menurut Martin Walters dan Jinny Johnson (2004), reptil merupakan hewan bertulang belakang (vertebrata) yang berdarah dingin serta memiliki sisik pada kulit dan meninggalkan telornya.
- Menurut John Woodward (2016), reptil merupakan vertebrata pertama yang hidup di darat dan memiliki sisik pada kulitnya setelah melewati evolusi yang panjang.
- Menurut Charlotte Uhlenbroek (2012), reptil merupakan hewan vertebrata yang memiliki tulang belakang, berdarah dingin, dan kulit yang dilapisi lapisan keras dan kering.
- Menurut Tom Jackson (2011), reptil merupakan hewan melata yang berevolusi dari dinosaurus yang telah punah dan berasal dari nenek moyang burung.

Berdasarkan pengertian menurut para ahli, dapat disimpulkan bahwa reptil adalah hewan berdarah dingin yang memiliki tulang belakang (vertebrata) serta dilapisi sisik tahan air yang keras pada kulit setelah melewati evolusi yang panjang dari nenek moyang burung.

2.3.2 Ordo Reptil

Menurut Charlotte Uhlenbroek (2012), reptil memiliki 4 ordo yang telah dikelompokkan menjadi beberapa bagian, yaitu:

- Ordo Chelonia, bangsa kura – kura.
- Ordo Rhynchocephalia, bangsa tuatara.
- Ordo Squamata, bangsa ular dan kadal.
- Ordo Crocodilia, bangsa buaya.

2.4 Ular

Pada bagian ini akan menjelaskan tentang pengertian dari ular, dan famili ular.

2.4.1 Pengertian Ular

Terdapat beberapa pengertian ular yang telah disampaikan oleh para ahli, diantaranya adalah sebagai berikut:

- Menurut David Alderton (2012), ular merupakan makhluk yang berevolusi dari kadal (leluhurnya) sejak dari 120 miliar tahun yang lalu dan memiliki anggota tubuh yang panjang dan tidak berkaki.

- Menurut Martin Walters dan Jinny Johnson (2004), ular merupakan hewan tidak berkaki yang telah berevolusi miliaran tahun yang lalu dari kadal yang memiliki 4 kaki. Ada sekitar 2400 spesies ular yang saat ini telah berhasil dikelompokkan dari berbagai wilayah kecuali Antartika.
- Menurut Derek Hall (2007), ular merupakan hewan yang berasal dari kelompok reptilia dalam ordo squamata yang memiliki nenek moyang berkaki dan kehilangan kaki selama perjalanan evolusi. Saat ini ular termasuk dalam kategori ophidia atau hewan tidak berkaki.
- Menurut Rhonda Klevansky (2015), ular merupakan predator pemakan daging yang memiliki tubuh panjang dan dilapisi sisik tahan air serta menelan mangsa secara utuh kedalam perutnya.

Berdasarkan pengertian menurut para ahli, dapat disimpulkan bahwa ular adalah reptilia pemakan daging yang memiliki tubuh panjang, sisik tahan air, dan telah berevolusi miliaran tahun dari leluhurnya, kadal, menjadi tidak berkaki (orphidia).

2.4.2 Famili Ular

Menurut Chris Mattison (2006), ada sebanyak 18 famili ular yang telah diklasifikasi dalam kelompok dengan spesies yang berbeda – beda. Tetapi dari 18 famili tersebut hanya ada 5 famili utama yaitu:

- Famili Boidae, kelompok ular boa dan anakonda (tidak berbisa).
- Famili Phytonidae, kelompok ular piton (tidak berbisa).
- Famili Colubridae, kelompok ular kolubrid (sebagian tidak berbisa).
- Famili Elapidae, kelompok ular laut, dan kobra (berbisa).

- Famili Viperidae, kelompok ular viper (berbisa).

2.5 Kolubrid

Pada bagian ini akan menjelaskan tentang pengertian dari kolubrid, dan ular jagung.

2.5.1 Pengertian Kolubrid

Terdapat beberapa pengertian kolubrid yang telah disampaikan oleh para ahli, diantaranya adalah sebagai berikut.

- Menurut Rhonda Klevanksy (2015), kolubrid merupakan ular yang tidak memiliki tulang pada paru - paru kiri dan pinggang serta sebagian besar dari mereka tidak berbisa.
- Menurut Chris Mattison (2006), kolubrid merupakan ular yang tidak memiliki korset pinggul, paru – paru kiri, dan tulang koronoid. Istilah pengelompokan kolubrid berasal dari garis evolusi.
- Menurut Derek Hall (2007), kolubrid merupakan ular terbanyak di dunia yang tersebar hampir seluruh dunia kecuali daerah dingin seperti antartika. Hampir setengah spesies ular di dunia masuk kelompok kolubrid dan ditemukan sebanyak 1760 jenis spesies kolubrid di dunia.
- Menurut RD. Bartlett dan Patricia Bartlett (1999), kolubrid merupakan kelompok ular yang memiliki persentasi sebanyak 78% dari total jenis ular di dunia. Bahkan di amerika, populasi ular yang termasuk keluarga kolubrid sebanyak 85%.

Berdasarkan pengertian menurut para ahli, dapat disimpulkan bahwa kolubrid adalah jenis pengelompokan ular terbanyak di dunia yang berasal dari garis evolusi yang tidak memiliki tulang pada paru - paru kiri, pinggang, koronoid dan sebagian besar tidak berbisa.

2.5.2 Ular Jagung

Dari banyaknya jenis kolubrid yang tersebar di dunia, ular jagung merupakan salah satu jenis ular dari keluarga kolubrid yang paling terkenal di kalangan pecinta reptil dunia. Menurut Don Soderberg (2006), ular jagung merupakan ular dari famili kolubrid yang berasal dari jenis ular tikus. Alasan dia disebut ular jagung adalah karena ia seringkali ditemukan di ladang jagung dan memakan tikus yang merusak jagung. Sehingga ular jagung memberi keuntungan kepada para petani yang mengolah ladang jagung tersebut.

Menurut RD. Bartlett dan Patricia Bartlett (1999), ular jagung memiliki beberapa deskripsi yaitu:

- Asal : Amerika.
- Nama latin : *elaphe guttata*.
- Nama lain : ular tikus merah (*red rat snake*).
- Tipe: terestial.
- Makanan: tikus, kadal, kodok.
- Berkembangbiak : ovipar (12-25 telur).
- Ukuran: 90cm – 180cm.
- Habitat: tumpukan kayu, lereng bukit batuan, dan lahan pertanian.

2.6 Android

Pada bagian ini akan menjelaskan tentang pengertian dari android, versi dari android, dan fitur dalam android.

2.6.1 Pengertian Android

Terdapat beberapa pengertian android yang telah disampaikan oleh para ahli, diantaranya adalah sebagai berikut:

- Menurut Mashuri (2015), android merupakan sistem operasi gratis dan open source yang berbasis linux untuk perangkat mobile.
- Menurut Andry (2011), android merupakan sistem operasi hasil modifikasi dari kernel linux yang berfungsi untuk perangkat mobile yang terdapat middleware (virtual machine) dan berbagai macam aplikasi utama.
- Menurut Safaat Harahap (2012), android merupakan sistem operasi berbasis Linux yang menyediakan platform terbuka (open source) bagi para pengembang bagi telepon seluler seperti telepon pintar dan komputer tablet.
- Menurut Supardi (2015), android merupakan sistem operasi yang memberi kesempatan kepada para pengembang untuk membuat aplikasi mereka sendiri di mobile device berbasis linux seperti smartphone dan tablet.

Berdasarkan pengertian menurut para ahli, dapat disimpulkan bahwa android adalah sistem operasi gratis yang terbuka bagi para pengembang (open source) untuk membuat aplikasi sendiri di mobile device pada telepon seluler yang berasal dari hasil modifikasi kernel linux serta memiliki middleware dan aplikasi utama yang berbasis linux.

2.6.2 Versi Android

Menurut Nazruddin Safaat H (2012), terdapat beberapa versi-versi android yaitu:

- Android versi 1.1 (*Bender*) yang dirilis pada tanggal 9 Maret 2009.
- Android versi 1.5 (*Cupcake*) yang dirilis pada bulan Mei 2009.
- Android versi 1.6 (*Donut*) yang dirilis pada bulan September 2009.
- Android versi 2.0 / 2.1 (*Eclair*) yang dirilis pada tanggal 3 Desember 2009.
- Android versi 2.2 (*Froyo: Frozen Yoghurt*) yang dirilis pada bulan Mei 2010.
- Android versi 2.3 (*Gingerbread*) yang dirilis pada bulan Desember 2010.
- Android versi 3.0/3.1 (*Honeycomb*) yang dirilis pada bulan Februari 2011.
- Android versi 4.0 (*Ice Cream Sandwich*) yang dirilis pada bulan November 2011.

2.6.3 Fitur Android

Menurut Andi (2013), terdapat beberapa fitur-fitur android yaitu:

- *Storage.*
- *Connectivity.*
- *Messaging.*
- *Web Browser.*
- *Media Support.*
- *Hardware Support.*
- *Multi Touch.*

- *Multitasking.*
- *Flash.*
- *Tethering.*

2.6.4 Arsitektur Android

Menurut Supardi (2015), terdapat beberapa arsitektur android yaitu:

- *Applications* dan *Widgets* merupakan layer yang memiliki hubungan hanya sebatas dengan aplikasinya.
- *Application Frameworks* merupakan layer open development platform yang ditawarkan oleh Android kepada orang yang membuat ataupun merancang suatu aplikasi.
- *Libraries* merupakan layer yang memiliki isi berupa fitur-fitur yang akan dijalankan pada Android.
- *Android Run Time* merupakan layer yang memiliki fungsi untuk menjalankan aplikasi Android.
- *Linux Kernel* merupakan layer yang menjadi pusat dari sistem operasi Android.

2.6.5 Kelebihan Android

Menurut Syafaat Nazruddin (2011), sistem operasi android memiliki kelebihan yaitu:

- *Complete Platform* karena sistem operasi android menyediakan tools yang berfungsi untuk membangun aplikasi yang dapat dikembangkan oleh para developer.

- *Open Source Platform* karena sistem operasi android bersifat terbuka sehingga mudah diperbaharui oleh para developer.
- *Free Platform* karena sistem operasi android dapat bebas dikembangkan oleh para developer, serta bebas diperdagangkan tanpa harus membayar lisensi.

2.7 Hereditas

Menurut Istamar Syamsuri (2007), orang yang pertama kali melakukan percobaan tentang pewarisan sifat adalah Gregor Johann Mendel (1822 - 1884). Ia adalah seorang biarawan di Brunn, Austria. Mendel memilih kacang kapri (pisum sativum) sebagai percobaan persilangan pertamanya dan mengamatnya selama 12 tahun. Hasilnya mendel menemukan ada 7 sifat berbeda yang mencolok pada tanaman kapri seperti pada tabel 2.1.

Tabel 2.1 Tanaman Kapri Mendel

Kapri Murni	Sifat Beda
Biji bulat	Biji keriput
Biji kuning	Biji hijau
Bentuk halus	Bentuk keriput
Warna hijau	Warna kuning
Bunga merah	Bunga putih
Bunga di ketiak daun	Bunga di ujung daun
Batang tinggi	Batang pendek

Alasan mendel memilih kapri sebagai percobaan persilangan pertama adalah sebagai berikut:

- Memiliki perbedaan pasangan sifat yang mencolok dan kontras.
- Dapat melakukan penyerbukan sendiri.

- Mudah untuk melakukan penyerbukan silang dengan cara mengambil serbuk sari dari satu kapri ke kapri lainnya.
- Waktu untuk menghasilkan keturunan sangat cepat sehingga dalam beberapa bulan sudah bisa diketahui hasilnya.
- Punya banyak keturunan.

Menurut Istamar Syamsuri (2007), dalam persilangan dari satu gen ke gen lainnya terdapat dua hasil berupa genotipe dan fenotipe. Genotipe merupakan sifat atau karakter bawaan yang telah ditentukan oleh gen keturunannya. Sehingga gen genotipe sulit untuk diprediksi karena bergantung pada kondisi lingkungan. Sedangkan fenotipe merupakan sifat yang dapat dilihat dari luar yang berasal dari paduan antara genotipe dan lingkungannya.

Dalam diagram persilangan, genotipe ditampilkan dalam bentuk simbol huruf, sedangkan fenotipe ditampilkan dalam bentuk kata. Contohnya, genotipe “K” untuk tumbuhan fenotipe berwarna “kuning”.

2.7.1 Hukum Mendel I

Menurut Irnaningtyas (2013), hukum mendel I (hukum segregasi) merupakan proses pemisahan antara pasangan alel secara bebas pada saat terjadi pembelahan meiosis dalam pembentukan gamet yang disertai dengan penurunan jumlah kromosom diploid menjadi haploid.

Pembuktian hukum mendel I ada pada penyilangan monohibrid. Monohibrid merupakan proses penyilangan yang dilakukan pada satu individu yang memiliki sifat yang berbeda dari satu pasangan alel.

Mendel mencoba melakukan persilangan kacang ercis berbunga warna ungu dengan kacang ercis berbunga warna putih. Ternyata hasil keturunan pertama semuanya berwarna ungu 100% karena warna ungu bersifat dominan dan warna putih bersifat resesif. Jika hasil keturunan pertama dipersilangkan lagi dengan sesamanya, maka akan muncul hasil keturunan kedua berupa 75% warna ungu dan 25% warna putih. Contoh dan cara perhitungan dapat dilihat di tabel 2.2.

Tabel 2.2 Hukum Mendel I

♀ \ ♂	U	u
U	UU Ungu	Uu Ungu
u	Uu Ungu	uu Putih

F1

P1 = ♀ UU (berbunga ungu) × ♂ uu (berbunga putih)

G = U u

F1 = 100% Uu (berbunga ungu)

P2 = ♀ UU (berbunga ungu) × ♂ uu (berbunga putih)

G2 = U,u U,u

F2

Rasio genotipe F2 = UU : Uu : uu

= 1 : 2 : 1

$$\begin{aligned}
 \text{Rasio fenotipe F2} &= \text{ungu : putih} \\
 &= 75\% : 25\% \\
 &= 3 : 1
 \end{aligned}$$

2.7.2 Hukum Mendel II

Menurut Irnaningtyas (2013), hukum mendel II (hukum assortasi) merupakan proses persilangan dimana setiap alel memiliki pasangan yang bebas dengan alel lainnya dengan syarat tidak sealel pada waktu pembentukan gamet. Pembuktian hukum mendel II ada pada penyilangan dihibrid. Dihibrid merupakan proses penyilangan antara dua individu yang memiliki sifat atau alel yang berbeda satu sama lain.

Mendel mencoba melakukan persilangan kacang ercis berbiji bulat dan berwarna kuning dengan kacang ercis berbiji keriput dan berwarna hijau. Kacang ercis berbiji bulat dan berwarna kuning bersifat dominan, sedangkan kacang ercis berbiji keriput dan berwarna hijau bersifat resesif. Jika keturunan pertama kacang ercis berbiji bulat dan berwarna kuning dipersilangkan dengan sesamanya, maka akan menghasilkan keturunan kedua, yaitu tanaman kacang ercis berbiji kuning, bulat hijau, keriput kuning, dan keriput hijau dengan keriput 9:3:3:1. Contoh dan cara perhitungan dapat dilihat di tabel 2.3.

Tabel 2.3 Hukum Mendel II

♂ ♀	BK	Bk	bK	bk
BK	BBKK Bulat Kuning	BBKk Bulat Kuning	BbKK Bulat Kuning	BbKk Bulat Kuning

Bk	BBKk Bulat Kuning	BBkk Bulat hijau	BbKk Bulat Kuning	Bbkk Bulat Hijau
bK	BbKK Bulat Kuning	BbKk Bulat Kuning	bbKK Keriput Kuning	bbKk Keriput Kuning
bk	BbKk Bulat Kuning	Bbkk Bulat Hijau	bbKk Keriput Kuning	bbkk Keriput Hijau

F1

P1 = ♀BBKK (biji bulat, warna kuning) × ♂bbkk (biji keriput, warna hijau)

G1 = BK bk

F1 = 100% BbKbk (biji bulat, warna kuning)

P2 = ♀BbKk (biji bulat, warna kuning) × ♂BbKk (biji keriput, warna hijau)

G2 = BK, Bk, Bk, bk BK, Bk, Bk, bk

F2

Rasio genotipe F2 = BBKK : BBKk : BbKK : BbKk : bbKK : Bbkk :

bbKk : bbkk

= 1 : 2 : 2 : 1 : 4 : 1 : 2 : 2 : 1

Rasio fenotipe F2 = bulat kuning : bulat hijau : keriput kuning :

keriput hijau

= 9 : 3 : 3 : 1

2.8 *System Development Life Cycle (SDLC)*

Pada bagian ini akan menjelaskan tentang pengertian dari SDLC, metodologi SDLC, dan tahapan SDLC.

2.8.1 *Pengertian System Development Life Cycle*

Terdapat beberapa pengertian SDLC yang telah disampaikan oleh para ahli, diantaranya adalah sebagai berikut:

- Menurut Kendall (2006), SDLC merupakan suatu tahapan untuk melakukan analisa dan membangun pada suatu rancangan sistem berdasarkan siklus kegiatan pengguna.
- Menurut Dennis, Wixom, dan Tegarden (2005), SDLC merupakan suatu proses untuk memahami jalannya sistem informasi yang mendukung kebutuhan bisnis, merancang dan membangun sistem, serta memberikannya kepada *user*.

Berdasarkan pengertian para ahli, dapat disimpulkan bahwa SDLC adalah suatu tahapan dalam melakukan analisa dan membangun suatu rancangan sistem informasi yang mendukung kebutuhan bisnis, merancang, dan membangun sistem berdasarkan siklus kegiatan *user*.

2.8.2 *Metodologi System Development Life Cycle*

Menurut Valacic, George & Hoffer (2012), SDLC memiliki beberapa metodologi untuk mengembangkan software, yaitu:

- *Waterfall*
- *Prototyping*

- *Rapid Application Development*
- *Interaction*

2.8.3 Tahapan System Development Life Cycle

Menurut Valacic, George & Hoffer (2012), SDLC memiliki beberapa tahapan, yaitu:

- *Planning.*
- *Analysis Selection.*
- *Design.*
- *Implementation.*
- *Operation.*

2.9 Waterfall

Pada bagian ini akan menjelaskan tentang pengertian dari waterfall, tahapan dalam waterfall, dan kelebihan dalam waterfall.

2.9.1 Pengertian Waterfall

Terdapat beberapa pengertian android yang telah disampaikan oleh para ahli, diantaranya adalah sebagai berikut:

- Menurut Presman (2010), waterfall merupakan model *classic* yang memiliki sifat sistematis, dan berurutan dalam proses membangun sebuah *software*.

- Menurut Galin (2004), waterfall merupakan metode yang menampilkan proses pengembangan secara keseluruhan dan dideskripsikan dalam urutan yang bersifat linear.

Berdasarkan pengertian para ahli, dapat disimpulkan bahwa waterfall adalah sebuah model dan metode proses perangkat lunak yang memiliki sifat sistematis dan berurutan dalam proses pengembangan sebuah *software* yang bersifat linear.

2.9.2 Tahapan Waterfall

Menurut Galin (2004), waterfall memiliki beberapa tahapan yaitu:

- *Requirements Definition*

Tahap ini merupakan proses analisis terhadap kebutuhan sistem dengan mengumpulkan berbagai macam data dan informasi sebagai landasan dalam dokumen *requirement*.

- *Analysis*

Tahap ini merupakan proses analisis yang dilakukan untuk menganalisa masalah apa yang menjadi landasan penelitian serta membangun desain dalam implementasi ke dalam sistem.

- *Design*

Tahap ini merupakan proses penerjemah sebagai syarat kebutuhan pada perancangan yang dapat dibangun sebelum membuat *coding* baik berupa struktur data, arsitektur perangkat lunak, representasi *interface*, maupun detail prosedural.

- *Coding*

Tahap ini merupakan proses menerjemah sebuah *design* menjadi bahasa yang dapat dipahami oleh komputer berupa pengkodean. Hal ini akan dibuat oleh para *programmer*.

- *System Tests*

Tahap ini merupakan proses pengujian terhadap sistem yang telah dirancang dan dibuat oleh *programmer*. Sehingga dengan pengujian ini akan mendeteksi kesalahan maupun perbaikan dalam sistem agar sistem tersebut dapat terus diperbaharui menjadi lebih baik.

- *Installation and Conversion*

Tahap ini merupakan proses menginstall suatu aplikasi yang telah matang dan sesuai dengan tujuan sehingga aplikasi tersebut siap digunakan oleh *user* dalam proses pemakaian.

- *Operation and Maintenance*

Tahap ini merupakan proses perawatan dan pemeliharaan aplikasi yang telah dirancang berupa terus memperbaharui informasi dan data yang dibutuhkan dalam aplikasi. Hal ini berupa peningkatan jasa sistem sebagai kebutuhan baru.

2.9.3 Kelebihan Waterfall

Menurut Rosa dan Salahudin (2011), waterfall memiliki beberapa kelebihan yaitu:

- Struktur pada tahap pengembangan sistem jelas dan terstruktur.
- Dokumentasi yang dihasilkan pada setiap tahap pengembangan terorganisir.

- Pengerjaan proyek terjadwal secara bertahap dan berurutan, serta mudah dikontrol.
- Proses yang paling sederhana, tahan lama, dan cocok untuk pengembangan perangkat lunak dengan spesifikasi yang tidak berubah-ubah.

2.10 Unified Modeling Language (UML)

Pada bagian ini akan menjelaskan pengertian UML, jenis UML, relasi UML, dan langkah UML.

2.10.1 Pengertian Unified Modeling Language

- Menurut Nugroho (2010), UML merupakan suatu bentuk bahasa pemodelan yang digunakan pada sistem atau perangkat lunak yang berorientasi objek sebagai pendukung para pengembang dalam menggambarkan alur dari sebuah sistem yang dibangun agar mudah dipelajari dan dipahami.
- Menurut Widodo (2011), UML merupakan bahasa pemodelan yang bersifat standar serta memiliki *semantic* dan sintak.

Berdasarkan pengertian menurut para ahli, dapat disimpulkan bahwa UML merupakan bahasa pemodelan standar yang digunakan pada sistem atau perangkat lunak dan berorientasi objek yang memiliki *semantic* dan sintak.

2.10.2 Jenis Unified Modeling Language

Menurut Nugroho (2010), UML memiliki beberapa jenis yaitu:

- *Use Case Diagram* merupakan grafis yang menggambarkan interaksi antara sistem, dan pengguna yang berfungsi untuk menggambarkan sekuensi langkah dari tiap interaksi.
- *Activity Diagram* merupakan diagram yang menggambarkan alur aktivitas dalam sebuah sistem yang sedang dibangun pada rangkaian aliran aktivitas proses bisnis dan *use case*.
- *Class Diagram* merupakan gambaran dari struktur objek sistem yang menunjukkan adanya interaksi antar kelas pada sistem.
- *Sequence Diagram* merupakan suatu grafis yang menggambarkan bagaimana objek saling berhubungan satu sama lain .
- *State Chart Diagram* merupakan gambaran pemodelan aktivitas objek khusus yang bersifat dinamis.

2.10.3 Relasi Unified Modeling Language

Menurut Nugroho (2010), UML memiliki beberapa relasi yaitu:

- *Dendency* merupakan hubungan yang terjadi perubahan pada elemen *independent* dan *dependent*.
- *Association* merupakan agregasi yang menampilkan hubungan suatu objek dengan bagian-bagiannya.
- *Generalization* merupakan hubungan antara objek anak yang berbagi perilaku dan struktur data dengan objek induknya.
- *Realization* merupakan hubungan realisasi yang dilakukan oleh suatu objek yang diwujudkan dari antara *elements* dengan *interface*, dan *use case* dengan *collaboration*.

2.10.4 Langkah Unified Modeling Language

Menurut Henderi (2008), UML memiliki beberapa langkah cara penggunaannya yaitu:

- Membuat sebuah daftar *business process* pada level yang tertinggi.
- Melakukan proses pemetaan terhadap *use case* pada setiap *business process*.
- Membuat sketsa *deployment diagram*.
- Mendefinisikan *requirement*, *security*, dan sebagai yang telah disediakan oleh sistem.
- Membuat *activity diagram* berdasarkan *use case diagram*.
- Mendefinisikan obyek level serta membuat *sequence* dan *collaboration*.
- Merancang *user interface model* yang menyediakan antar muka bagi pengguna.
- Membuat *class diagram* berdasarkan model-model yang telah dirancang sebelumnya.
- Membuat *component diagram* dan mendefinisikan pengujian integrasi pada setiap komponen.
- Memperbaharui *deployment diagram* yang sudah dirancang.
- Membangun sebuah sistem.