BAB II
TINJAUAN PUSTAKA

2.1 Penelitian Sebelumnya

Penelitian ini bertujuan untuk membantu dalam penyimpanan berkas digital dalam investigasi suatu kasus kejahatan agar berkas tersebut terjamin keamanannya dan tidak disalahgunakan oleh pihak tertentu dengan merubah bahkan menghilangkan berkas bukti digital. Adapun beberapa referensi yang berhubungan dengan penelitian ini, yaitu:

1. Cahyo Handoko (2017) Referensi pertama, mendeskripsikan alat bukti digital pada perkara *cybercrime*, mengetahui kedudukan alat bukti digital pada perkara *cybercrime*. Teori yang digunakan dalam penelitian ini merupakan teori hukum progresif dengan metode penelitian yuridis empiris dengan menggunakan analisis data secara kualitatif, jenis penelitiannya merupakan penelitian deskriptif[1].
2. Agus Wibowo dan Ariana Azimah (2016) Referensi kedua, melakukan perancangan dan implementasi sistem informasi penjaminan mutu perguruan tinggi. Tahapan dalam perancangan dengan metode *throwaway protoyping* yaitu melakukan fase *planning*, analisis, desain menentukan kebutuhan, desain arsitektur, desain arsitektur sistem, mengevaluasi arsitektur, dan implementasi untuk pengembangan *prototype*[4].
3. Alicia Sinsuw dan Xaverius Najoan (2013) Referensi ketiga, merancang sistem informasi akademik berbasis android, yang dapat diakses melalui perangkat bergerak (*mobile)* untuk memudahkan akses yang lebih fleksibel. *Input* sistem mengacu kepada sistem informasi akademik yang telah diimplementasikan di Universitas Sam Ratulangi. Keluaran dari sistem informasi ini dapat memberikan informasi yang relevan dan *up to date* bagi *stakeholder*[5].
4. Afghan Amar Pradipta, Yuli Adam Prasetyo, dan Nia Ambarsari (2015) Referensi keempat, *web e-commerce* Bojana Sari yang telah dikembangkan

menggunakan metode *prototype* memiliki fitur yaitu pengelolaan produk , sistem pemesanan, sistem pembayaran, dan pelacakan status. Dalam fitur pengelolaan produk, perusahaan dapat dengan mudah menyampaikan informasi menegenai produk kepada pelanggan. Pelanggan dapat dengan mudah memperoleh informasi tersebut tann pa terhalang jarak dan waktu. Proses pemesanan dan validasi pembayaran dilakukan dengan fitur sistem pemesanan dan fitur pembayaran. Dengan adanya fitur tersebut, maka *webi e-commerce* yang telah dikembangkan mampu memenuhi kebutuhan perusahaan[3].

5. Neni Purwati dan Hendra Kurniawan (2015) Referensi Kelima, dari perkembangan pada penelitian ini ialah proses pengecekan judul skripsi atau tugas akhir dimodelkan dengan aplikasi *dekstop* dengan melibatkan jurusan sebagai *checker* dan BAAK sebagai *publisher* SK. Aplikasi yang dikembangkan dapat dijadikan alternatif dalam proses pengecekan judul skripsi atau tugas akhir di Institut Informatika dan Bisnis Darmajaya [6].

Tabel 2 1 Tinjauan Pustaka

No.	Judul, Nama Penulis, Tahun	Isi	Perbandingan
1.	Kedudukan Alat Bukti Digital dalam Pembuktian *Cybercrime* di Pengadilan, Cahyo Handoko (2017)	Pada jurnal penelitian ini membahas tentang mendeskripsikan profil alat bukti digital pada perkara *cybercrime.*	Perbedaan penelitian ini yaitu pada metode dan studi kasus, serta penelitian ini tidak menggunakan *tools*.
2.	*System Usability Scale* Antarmuka Palembang *Guide* Sebagai Media Pendukung Asian Games XVIII (2017)	Pada jurnal penelitian ini membahas tentang untuk menjamin perangkat lunak palembang *guide* layak untuk digunakan oleh pengguna akhir maka dilakukan pengujian antarmuka menggunakan instrumen *system usability scale*.	Perbedaan penelitian ini yaitu dalam studi kasus.
3.	Prinsip dan Prosedur Dasar Penanganan Bukti dalam *Computer Crime* dan *Compute Related Crime* (2014)	Prinsip dasar dan prosedur digital forensik memegang peranan penting untuk mengarahkan digital forensik untuk tetap berada pada jalur yang benar. Forensik digital mempunyai standar dalam proses penanganan barang bukti.	Perbedaan dalam penelitian ini yaitu studi kasus yang digunakan, metode serta penelitian ini tidak menggunakan *tools*.
4.	Kasus *Cybercrime* di Indonesia, Dista Amalia Arifah (2011)	Pada jurnal ini membahas tentang *cybercrime*, cara menanganinya serta solusi dalam semua hambatan bagi penyelidikan.	Perbedaan dalam penelitian ini adalah metode dan studi kasusnya. Penelitian sebelumnya tidak mengguanakan *tools.*
5.	*Penetration Testing* pada *Domain* UII.AC.ID Menggunakan OWASP 10 (2018)	Pada jurnal ini membahas tentang melakukan analisis terhadap sistem dan jaringan yang terdapat pada UI dari perspektif luar atau jaringan publik.	Perbedaan dalam penelitian ini yaitu *tools* yang digunakan serta studi kasusnya.
6.	Pengembangan *Web e-commerce* Bojana Sari Menggunakan *Prototype (2015)*	Pada jurnal penelitian ini membahas tentang bagaimana mengembangkan sebuah *web e-commerce* untuk bisnis.	Perbedaan dalam penelitian ini yaitu terdapat pada studi kasusnya dan *tools* yang digunakan.

No.	Judul, Nama Penulis, Tahun	Isi	Perbandingan
7.	Pengujian Sistem Informasi Pengelolaan Pelatihan Kerja UPT BLK Kabupaten Kudus dengan Metode *White Box Testing* (2017)	Pada jurnal penelitian ini membahas tentang eksperimen penerapan pengujian unit pada sistem menggunakan metode *white box* yang telah dibangun pada penelitian sebelumnya.	Perbedaan dalam penelitian ini yaitu dalam studi kasusnya.

2.2 Dasar Teori

2.2.1 *Cybercrime*

Cybercrime atau kejahatan dunia maya adalah istilah yang mengacu kepada aktivitas kejahatan dengan komputer atau jaringan komputer menjadi alat, sasaran, atau tempat terjadinya kejahatan. Walaupun kejahatan dunia maya atau *cybercrime* umumnya mengacu kepada aktivitas kejahatan dengan komputer atau jaringan komputer sebagai unsur utamanya, istilah ini juga digunakan untuk kegiatan kejahatan tradisional dimana komputer atau jaringan komputer digunakan untuk mempermudah atau memungkinkan kejahatan itu terjadi.

Cybercrime adalah suatu bentuk kejahatan virtual dengan memanfaatkan media komputer yang terhubung ke internet, dan mengeksploitasi komputer lain yang terhubung dengan internet juga. Adanya lubang-lubang keamanan pada sistem operasi menyebabkan kelemahan dan terbukanya lubang yang dapat digunakan para *hacker*, *cracker,* dan *script kiddies* untuk menyusup kedalam komputer tersebut. *Cybercrime* juga bisa didefenisikan sebagai segala macam penggunaan jaringan komputer untuk tujuan kriminal atau kriminal berteknologi tinggi dengan menyalahgunakan kemudahan teknologi digital [2].

Dari beberapa penjelasan diatas dapat disimpulkan bahwa *cybercrime* adalah tindakan yang melanggar hukum dengan menggunakan media teknologi dalam perbuatannya yang dapat merugikan orang lain. *Cybercrime* dapat diklasifikasikan menjadi tiga, yaitu *cyberpiracy*, *cybertresspass,* dan *cybervandalism. Cyberpiracy* adalah penggunaan teknologi komputer untuk mencetak ulang *software* atau informasi lalu mendistribusikan informasi atau *software* tersebut melalui teknologi komputer. *Cybertresspass* adalah penggunaan teknologi komputer untuk meningkatkan akses pada sistem komputer suatu organisasi atau individu. *Cybervandalism* penggunaan teknologi komputer untuk membuat program yang mengganggu proses transmisi elektronik, dan menghancurkan data komputer. Jenis-jenis *cybercrime* berdasarkan aktivitasnya, sebagai berikut:

1. *Unauthorized access to Computer system and Service*
2. *Illegal contents*

3. *Data Forgery*
4. *Cyber espionage*
5. *Cyber Sabotage and exortion*
6. *Offense against Intellectual Property*
7. *Infringements of Privacy*
8. *Cracking*
9. *Carding*

2.2.2 Bukti Digital

Bukti digital adalah sebuah data yang disimpan atau dikirimkan menggunakan komputer yang dapat mendukung atau menyangkal sebuah pelanggaran tertentu, atau bisa juga disebut sebagai petunjuk yang mengarahkan kepada elemen-elemen penting berkaitan dengan sebuah pelanggaran. Bukti digital juga bisa disebut sebagai abstraksi dari beberapa objek digital atau kejadian. Ketika seseorang mengoperasikan komputer untuk melakukan berbagai hal seperti mengirim *email*, atau kegiatan lainnya maka kegitatan itu akan menghasilkan jejak data yang dapat memberikan sebagian gambaran dari kejadian yang sudah terjadi sebelumnya[7].

Undang-Undang Nomor 11 Tahun 2008 Tentang Informasi dan Transaksi Elektronik atau UU ITE yang merupakan pedoman hukum *cyber* di Indonesia ternyata tidak mencantumkan penjelasan tentang bukti digital. Namun, terdapat dua istilah yang mirip mengenai bukti digital yaitu informasi elektronik dan dokumen elektronik. Dalam Pasal 1 Butir 1 UU ITE disebutkan bahwa informasi elektronik adalah satu atau sekumpulan data elektronik, termasuk tetapi tidak terbatas pada tulisan, suara, gambar, peta, rancangan, foto, *Electronic Data Interchange* (EDI), *electronik mail* (surat elektronik), telegram, teleks, telecopy atau sejenisnya, huruf, tanda, angka, kode akses, simbol, atau perforasi yang telah diolah yang memiliki arti atau dapat dipahami oleh orang yang mampu memahaminya. Sedangkan dalam Pasal 1 Butir 4 UU ITE, menjelaskan bahwa dokumen elektronik adalah setiap informasi elektronik yang dibuat, diteruskan, dikirimkan, diterima, atau disimpan dalam bentuk analog, digital,

elektromagnetik, optikal, atau sejenisnya yang dapat dilihat, ditampilkan, dan atau didengar melalui komputer atau sistem elektronik, termasuk tetapi tidak terbatas pada tulisan, suara, gambar, peta, rancangan, foto atau sejenisnya, huruf, tanda, angka, kode akses, simbol, atau perforasi yang telah diolah yang memiliki arti atau dapat dipahami oleh orang yang mampu memahaminya.[8]

Dari pengertian diatas dapat dirumuskan bahwa bukti digital merupakan kumpulan dari informasi dan dokumen elektronik yang tersimpan dalam perangkat elektronik serta bukti yang sah dan dapat dipertanggung jawabkan di hadapan hukum. Dalam setiap kasus kejahatan teknologi informasi, maka akan ditemukan berbagai barang bukti yang terdapat pada Tempat Kejadian Perkara (TKP). Barang bukti tersebut kemudian akan dijadikan bukti digital yang didalamnya terdapat rekam jejak pelaku selama *cybercrime* yang dilakukan. Bukti digital yang didalamnya terdapat beberapa elemen penting yang dijadikan sebagai bukti yang dapat mendukung atau menjerumuskan pelaku didalam persidangan.

Bukti digital kini telah diakui di Indonesia sesuai dengan Undang-Undang Republik Indonesia Nomor 11 Tahun 2008 Tentang Informasi Dan Transaksi Elektronik, bahwa informasi atau dokumen elektronik merupakan alat bukti hukum yang sah. Abdullah (2007) mengemukakan bahwa ada beberapa aturan standar agar bukti-bukti digital dapat diterima dalam proses peradilan. Terdapat 5 karakteristik bukti digital yaitu:

1. *Admissible* (Layak dan dapat diterima)
2. *Authentic* (Asli)
3. *Complete* (Lengkap)
4. *Reliable* (Handal)
5. *Believable* (Terpercaya)

Metode forensik TI memiliki empat elemen forensik yang menjadi kunci dalam proses pengungkapan bukti digital. Empat elemen tersebut ialah:

1. Identifikasi bukti digital, pada tahapan ini perlu dilakukan identifikasi dimana bukti itu bersumber, dimana bukti itu disimpan, dan bagaimana penyimpanannya untuk mempermudah penyelidikan. Pihak yang perlu

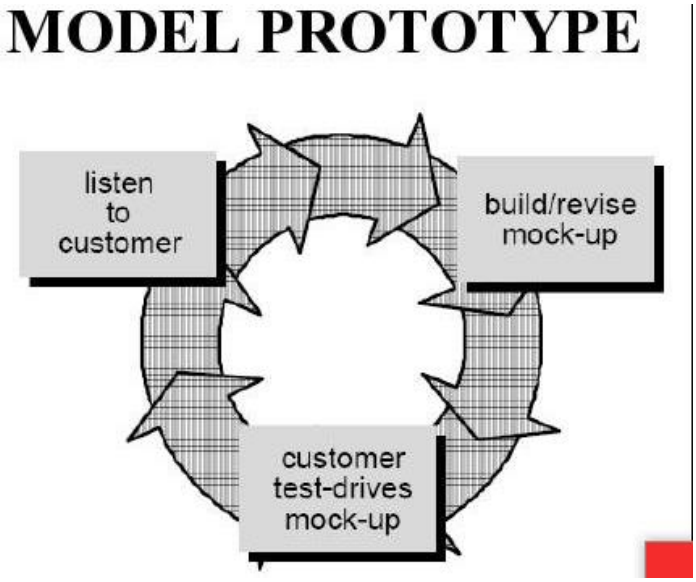
dilibatkan dalam proses ini adalah para petugas keamanan, penelaah bukti, dan teknisi khusus.

2. Penyimpanan bukti digital, bentuk, isi, makna dari bukti digital hendaknya disimpan dalam tempat yang steril. *Copy* data secara *bitstreamimage.* Teknik pengkopian ini menggunakan teknik komputasi CRC, teknik ini umumnya di istilahkan dengan *cloning disk* atau *ghosting.*
3. Analisa bukti digital , barang bukti yang telah didapatkan perlu dikembangkan kembali kedalam sejumlah skenario yang berhubungan dengan tindak pengusutan sehingga didapat hasil analisa antara lain siapa yang telah melakukan, apa yang telah dilakukan, dan waktu melakukan. Secara umum, setiap data yang ditemukan dalam sebuah sistem komputer sebenarnya adalah potensi informasi yang belum diolah, sehingga keberadaannya memiliki sifat yang cukup penting. Dalam proses analisa forensik terkhusus pada hardisk dapat dilakukan terhadap semua jenis sitem operasi yang diinginkan.
4. Presentasi bukti digital , kesimpulan akan didapatkan ketika semua tahapan telah dilalui, terlepas dari ukuran obyektifitas yang didapatkan, atau standar kebenaran yang diperoleh, minimal bahan-bahan inilah nanti yang akan dijadikan bukti untu mengungkap sebuah kasus yang berkaitan dengan kejahatan komputer. Selanjutnya bukti-bukti digital akan diuji otentifikasi dan dikorelasikan dengan kasus yang ada. Pada tahapan ini semua proses yang telah dilakukan sebelumnya akan diurai kebenarannya serta dibuktikan kepada hakim untuk mengungkap data dan informasi kejadian.

2.2.3 *Prototype*

Metode *prototype* adalah metode pengembangan perangkat lunak yang memodelkan dari sistem kerja suatu perangkat lunak yang belum lengkap dari pihak *user*. Model *prototype* digunakan untuk mendemonstrasikan beberapa konsep, percobaan rancangan, dan menemukan banyak masalah dan solusi yang memungkinkan. Sistem *prototype* memperbolehkan pengguna untuk mengetahui bagaimana sistem berjalan dengan baik, dengan metode *prototype* pengembang dan pengguna dapat saling berinteraksi selama proses pembuatan sistem [9].

Penggunaan metode *prototyping* dalam penelitian ini bertujuan agar peneliti mendapatkan gambaran aplikasi yang akan dibangun melalui tahap pembangunan aplikasi *prototype* terlebih dahulu yang akan dievaluasi oleh *user*.



Gambar 1.1 Model Prototype

Gambar 2.1 menjelaskan bahwa metode *prototype* dimulai dengan mendengarkan kebutuhan dan masukan dari pengguna. Pengembang dan pengguna bertemu bersama-sama menentukan tujuan keseluruhan untuk perangkat lunak dan mengidentifikasi apapun persyaratan yang diperlukan. Kemudian, pengembang membuat sebuah gambaran tentang aplikasi yang selanjutnya dapat diprensentasikan kepada pelanggan. Gambaran tersebut berfokus pada representasi aspek aplikasi yang diinginkan oleh *user*. Beberapa keunggulan dalam menggunakan metode *prototype*, antara lain:

1. Pengembang sistem dan pengguna saling berkomunikasi khususnya dalam hal penyamaan persepsi terhadap pemodelan sistem yang akan menjadi dasar pengembangan sistem operasionalnya.
2. Pengguna ikut terlibat secara aktif dan berpartisipasi dalam menentukan model sistem sehingga pengguna akan puas karena sistem yang dibuat sesuai dengan keinginan dan harapannya
3. Sistem yang dibangun memiliki kualitas yang diinginkan karena sesuai dengan kebutuhan yang ada.

2.2.4 PHP

PHP adalah bahasa *server-side-scripting* yang menyatu dengan HTML untuk membuat halaman *web* yang dinamis. PHP adalah bahasa *server-side–scripting*, maka sintaks dan perintah PHP akan dieksekusi diserver kemudian hasilnya akan dikirimkan ke *browser* dengan format HTML. Dengan demikian, kode program yang ditulis dalam PHP tidak akan terlihat oleh user sehingga keamanan halaman *web* lebih terjamin. PHP dirancang untuk membuat halaman web yang dinamis, yaitu halaman *web* yang dapat membentuk suatu tampilan berdasarkan permintaan terkini, seperti menampilkan isi basis data ke halaman *web*.

PHP atau singkatan dari *Personal Home Page* merupakan bahasa *script* yang tertanam dalam HTML untuk dieksekusi bersifat *server side.* PHP termasuk dalam *open source product*, sehingga *source code* PHP dapat diubah dan didistribusikan secara bebas. Versi terbaru PHP dapat diunduh secara gratis melalui situs resmi PHP.

PHP juga dapat berjalan terhadap berbagai *web* seperti IIS (*Internet Information Sever*), PWS (*Personal Web Server*), Apache, Xitami. PHP juga mampu berjalan dibanyak sistem operasi yang beredar saat ini, diantaranya Sistem Operasi Microsoft Windows, Linux, Mac Os, Solaris. PHP dapat dibangun sebagai modul *web* server Apache dan sebagai *binary* yang dapat berjalan sebagai CGI (*Common Gateway Interface*).

Salah satu keunggulan yang dimiliki PHP adalah kemampuannya untuk melakukan koneksi ke berbagai macam software sistem manajemen basis data atau *Database Management System* (DBMS), sehingga dapat menciptakan suatu halaman *web* dinamis. PHP mempunyai konektivitas yang baik dengan beberapa DBMS seperti Oracle, Sybase, mSQL, MySQL, Microsoft SQL Server, Solid, PostgresSQL, Adabas, FilePRo, Velocis, dBase, Unix dbm, dan tidak terkecuali semua database ber-*interface* ODBC. PHP diciptakan oleh Rasmus Lerdorf Pertama kali tahun 1994[10].

2.2.5 *Database* MySQL

MySQL merupakan salah jenis database server yang sangat terkenal dan banyak digunakan untuk membangun aplikasi web yang menggunakan database sebagai sumber dan pengolahan datanya.

MySQL dikembangkan oleh Perusahaan Swedia bernama MySQL AB yang pada saat ini bernama Tcx Data Konsult AB sekitar tahun 1994-1995, namun cikal bakal kodenya sudah ada sejak tahun 1979. Awalnya Tcx merupakan perusahaan pengembang software dan konsultan *database*, dan saat ini MySQL sudah diambil alih oleh Oracle Corp.

Kepopuleran MySQL antara lain karena MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses *database*nya sehingga mudah untuk digunakan, kinerja *query* cepat, dan mencukupi untuk kebutuhan *database* perusahaan-perusahaan yang berskala kecil sampai menengah, MySQL juga bersifat *open source* (tidak berbayar).

MySQL merupakan *database* yang pertama kali didukung oleh bahasa pemrograman *script* untuk internet (PHP dan Perl). MySQL dan PHP dianggap sebagai pasangan *software* pembangun aplikasi *web* yang ideal. MySQL lebih sering digunakan untuk membangun aplikasi berbasis *web*, umumnya pengembangan aplikasinya menggunakan bahasa pemrograman *script* PHP[11].

2.2.6 HTML

Hyper Text Markup Language (HTML) adalah sebuah bahasa markah yang digunakan untuk membuat sebuah halaman *web*, menampilkan berbagai informasi didalam sebuah penjelajah *web* Internet dan pemformatan hiperteks sederhana yang ditulis dalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegerasi. Berikut beberapa referensi pengertian HTML menurut para ahli.

HTML adalah bahasa pemfomatan teks untuk dokumen-dokumen pada jaringan komputer yang sering disebut sebagai *world wide web*. Sedangkan menurut Arief, HTML merupakan salah satu format yang digunakan dalam pembuatan dokumen atau aplikasi yang berjalan di halaman *web*, dan menurut

Suyanto yang dikutip oleh Agustinus Afrano Amran dalam penelitiannya ,HTML itu adalah bahasa yang digunakan untuk menulis halaman *web,* biasanya menggunakan extensi .htm, .html atau .shtml.

Dan menurut Jubilee Enterprise HTML berawal dari bahasa SGML (*Standard Generalized Markup Language*) yang penulisannya disederhanakan. HTML dapat dibaca oleh berbagai macam *flatform*. HTML juga merupakan bahasa pemrograman yang *fleksible* dan dapat digabungkan dengan bahasa pemrograman lain seperti PHP, ASP, JSP,JavaScript[12].

2.2.7 Algoritma AES

Advanced Encryption Standard atau biasa disebut AES merupakan algoritma kriptografi yang digunakan untuk mengamankan data. Algoritma AES adalah blok *ciphertext* simetrik yang dapat mengenkripsi dan deskripsi informasi. Enkripsi merubah data yang tidak dapat lagi dibaca disebut *ciphertext*. Sedangkan deskripsi adalah merubah *ciphertext* data menjadi bentuk semula yang kita kenal sebagai *plaintext*. Algoritma AES memiliki tiga pilihan kunci, yaitu tipe: AES-128, AES-192, dan AES-256. Masing-masing tipe kunci internal yang berbeda yaitu *round key* untuk setiap proses putaran. Proses putaran enkripsi AES-128 dikerjakan sebanyak 10 kali (a=10).

Proses Enkripsi Algoritma AES terdiri dari 4 jenis transformasi bytes, yaitu *SubBytes, ShiftRows, Mixcolumns,* dan *AddroundKey*. Pada awal proses dilakukan enkripsi , *input* yang telah di *copy* kedalam *state* akan mengalami transformasi *byte AddroundKey.* Setelah itu, *state* akan mengalami *SubBytes, ShiftRows, MixColumns* dan *AddRoundKey* secara berulang-ulang sebanyak Nr. Proses ini dalam algoritma dapat disebut dengan *round function. Round* yang terakhir agak berbeda dengan *round-round* sebelumnya dimana pada *round* terakhir, *state* tidak mengalami transformasi *MixColumns*[15].

2.2.8 SSL

SSL kepanjangan dari *Secure Socket Layer* adalah sebuah protokol yangbertujuan untuk memberikan keamanan sebuah website. *Secure Sockets*

Layer adalah sebagai pelindung otentikasi dan *traffic* data dari server ke klien maupun sebaliknya. Aplikasi yang digunakan untuk SSL adalah menggunakan *OpenSSL*[16].

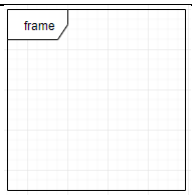
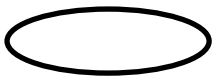
2.2.9 UML

Unified Modeling Languange atau biasa disebut UML adalah sebuah teknik pengembangan sistem yang menggunakan bahasa grafis sebagai alat untuk dokumentasi dan melakukan spesifikasi pada sistem. UML memiliki banyak diagram yang digunakan untuk melakukan pemodelan data maupun sistem seperti *Use Case Diagram, Activity Diagram, Class Diagram,* dan *Sequence Diagram*[17].

1. *Use Case Diagram*

   Use case diagram yaitu diagram yang digunakan untuk menggambarkan hubungan antara sistem dengan actor. Diagram ini hanya menggambarkan secara global sehingga komponen-komponen yang digunakan sedikit. Komponen-komponen yang digunakan pada *use case diagram* dapat dilihat pada tabel di bawah ini[18].

Tabel 2.2 Tabel Komponen-Komponen *Use Case Diagram*

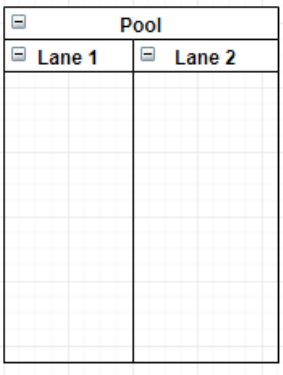
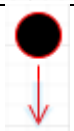
Notasi	Komponen	Keterangan
	Sistem	Batasan-batasan proses yang sudah kita deskripsikan dalam sebuah sistem.
	Aktor	Elemen yang menjadi pemicu sistem. Aktor bisa berupa orang, mesin maupun sistem lain yang berinteraksi dengan *use case*
	*Use case*	Potongan proses yang merupakan bagian dari sistem
	Association	Menggambarkan interaksi antara *use case* dan aktor
	*Dependency*	Relasi antara dua *use case* dengan dua tipe yaitu *include* dan *extends*. *Include* menghubungkan dua *use*

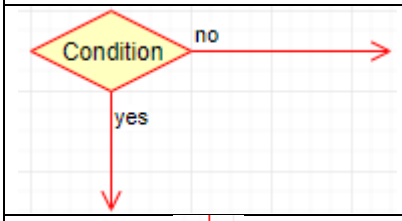
Notasi	Komponen	Keterangan
		case dimana satu *use case* membutuhkan *use case* lain. Sedangkan *extends* menghubungkan dua *use case* dimana suatu *use case* terkadang memanggil *use case* lain, tergantung pada kondisi
	*Generalization*	Pewarisan antara dua aktor atau *use case* dimana salah satu aktor atau *use case* mewarisi *properties* ke aktor atau *use case* yang lain.

2. *Activity Diagram*

Activity diagram yaitu diagram yang digunakan untuk menggambarkan alur kerja (aktivitas) pada *use case* (proses), logika, proses bisnis dan hubungan antara aktor dengan alur-alur kerja *use case*. Komponen-komponen yang digunakan pada *activity diagram* dapat dilihat pada tabel di bawah ini.

Tabel 2.3 Komponen-Komponen *Activity Diagram*

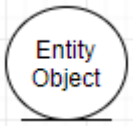
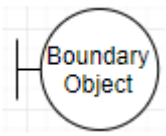
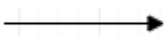
Notasi	Komponen	Keterangan
	*Swimlines*	Memisahkan antara aktor dan sistem ataupun antara aktor satu dengan yang lain atau antara sistem yang satu dengan yang lain
	*Start point*	Memulai *activity diagram*
	*Activities*	Untuk menggambarkan aktivitas
	*Transitions*	Transisi dari elemen yang satu ke elemen yang lain

Notasi	Komponen	Keterangan
	*Decisions*	Percabangan logika dengan dua buah pilihan. Sering dijumpai pada *flowchart*.
	*End point*	Mengakhiri *activity diagram*

3. *Sequence Diagram*

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Komponen-komponen yang digunakan dalam *sequence diagram* dapat dilihat pada tabel di bawah ini.

Tabel 2.4 Komponen-Komponen *Sequence Diagram*

Notasi	Komponen	Keterangan
	Aktor	Elemen yang menjadi pemicu sistem. Aktor bisa berupa orang, mesin maupun sistem lain yang berinteraksi dengan *use case*
	*Entity class*	Kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem
	*Boundary class*	Kumpulan kelas yang menjadi *interfaces* atau interaksi antara satu atau lebih aktor dengan sistem
	*Control class*	Berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya kalkulasi dan aturan bisnis yang melibatkan berbagai objek
	*Message*	Mengirim pesan antar *class*
	*Activation*	Mewakili eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivasi sebuah operasi

Notasi	Komponen	Keterangan
	*Lifeline*	Garis titik-titik yang terhubung dengan objek dimana sepanjang *lifeline* terdapat *activation*

2.2.10 OWASP TOP 10

OWASP TOP 10 atau yang biasa disebut Owasp 10 adalah sebuah daftar yang dirilis oleh komunitas OWASP yang berisikan 10 daftar teratas celah keamanan yang dapat mengancam keamanan suatu *website*. Daftar ini terus berkembang dan berubah-ubah mengikuti perkembangan teknologi *website* yang terus berkembang. OWASP 10 pertama kali dirilis tahun 2003 lalu *update minor* pada tahun 2004 dan 2007 dan 2010. OWASP 10 sendiri dibuat dengan tujuan untuk meningkatkan kesadaran tentang keamanan dengan mengidentifikasi beberapa resiko celah keamanan yang sering dihadapi atau ditemui dalam banyak kasus seperti pada gambar 2.2 dibawah ini[19].

***OWASP TOP* 10 – 2013**	***OWASP TOP* 10 – 2017**
A1 – Injeksi	A1 – Injeksi
A2 – Otentikasi dan manajemen sesi yang buruk	A2 – Otentikasi yang buruk
A3 – *Cross-Site Scripting (XSS)*	A3 – Data sensitif yang terekspos
A4 – Referensi obyek langsung yang tidak aman	A4 – *XML External Entities (XXE)*
A5 – Kesalahan konfigurasi keamanan	A5 – Akses kontrol yang buruk
A6 – Data sensitif yang terekspos	A6 – Kesalahan konfigurasi keamanan
A7 – Kehilangan fungsi kontrol tingkatan akses	A7 – *Cross-Site Scripting (XSS)*
A8 – *Cross-Site Request Forgery (CSRF)*	A8 – Deserialisasi yang tidak aman
A9 – Menggunakan komponen rentan yang diketahui	A9 – Menggunakan komponen rentan yang diketahui
A10 – Redireksi dan *Forward* yang tidak tervalidasi	A10 – Pencatatan dan pemantauan yang tidak cukup

Gambar 2.2 OWASP TOP 10 tahun 2013 dengan 2017

Sumber : (OWASP 2017)

Gambar diatas adalah contoh OWASP 10 yang dirilis oleh organisasi OWASP yang berisikan 10 celah keamanan yang sering ditemukan pada tahun 2013 dan 2017.

1. SQL *Injection*

   SQL (*Structured Query Language*) *Injection* merupakan suatu teknik peretasan yang memungkinkan penyerang mendapatkan akses yang tidak sah kedalam *database* kemudian menyerang atau mengubah data-data yang berada didalam *database*. Dalam teknik *SQL Injection*, diperlukan karakter-karakter khusus untuk memanipulasi perintah SQL. Karakter-karakter tersebut yakni *double minus* (--) dan *Union*. *Double minus* merupakan tanda akhir suatu perintah SQL, sehingga perintah yang berada dibelakang *double minus* tidak akan terbaca dan tereksekusi oleh MySQL. *Union* merupakan perintah untuk menggabungkan dua atau lebih perintah SQL. *Union* sendiri digunakan untuk memberikan perintah *error* dalam *query*[20].

2. *Sensitive Data Exposure*

   Banyak aplikasi web tidak benar dalam melindungi data yang sensitive, seperti kartu kredit, id pajak dan pembuktian surat-surat berharga. Penyerang dapat mencuri atau memodifikasi data yang lemah dilindungi tersebut atau melakukan pencurian identitas, penipuan kartu kredit, atau kejahatan lainnya. Data *sensitive* layak mendapatkan perlindungan ekstra seperti enkripsi[20].

3. HTTP Respon

   Request dari klien ditangani oleh server dan diresponnya secara tepat. Dalam direspon, server mengirimkan kembali serangkaian komponen pesan yang dapat dikategorikan menjadi:

   a. Kode Respon adalah sebuah kode angka yang sesuai dengan respon yang bersangkutan. Kode respon merupakan bagian awal dari respon server dan menetapkan nada untuk mengingatkan interaksi tersebut. Server merespon dalam salah satu 4 (empat) cara ini: *Success, Redirection, Client Error,* atau *Server Error*. Tabel 2.4 menunjukkan kode-kode respon umum yang sering digunakan[21].

Tabel 2.5 Kode Respon

Kode Respon	**Keterangan**
Success 2xx	
200 OK	Request Berhasil dijalankan
Redirection 3xx	
301 Moved Permanently	Sumber-sumber request diberi URL permanen dan ditempatkan pada field location. Kode respon ini berkata, " saya telah dipindahkan, ikuti saya ketempat yang baru"
302 Moved Permanently	Sumber-sumber request telah diberi URL temporer yang baru dan ditempatkan pada field location. Kode respon ini berkata, " saya telah dipindahkan, ikuti saya ketempat sementaraku yang baru, tetapi jangan bergantung pada saya ika ingin disini dalam waktu lama."
Client Error 4xx	
400 Bad Request	Request tak dimengerti oleh server
401 Unauthorized	Sumber-sumber yang diminta
	memerlukan pembuktian keaslian user, biasanya dalam bentuk pembuktian keaslian basic atau yang sejenis.
403 Forbidden	Server memahami request tetapi menolak untuk merespon. Khususnya sewaktu metode GET digunakan untuk menerima respon ini, sedikit atau tak ada informasi lebih jauh lagi. Tetapi, sewaktu metode HEAD dipakai , beberapa server akan memberi informasi rinci lebih lengkap mengenai mengapa kondisi seperti ini muncul.
404 Not Found	Sumber-sumber yang diminta tidak ditemukan.
Server Error 5xx	
500 Internal Server Error	Server menemukan kesalahan internal (Internal error) dalam memproses request.
501 Not Implemented	Server tidak mendukung request yang bersangkutan

Kode Respon	Keterangan
502 Bad Gateway	Server menerima respon cacat dari server utama sewaktu ia mengirim resource yang diminta. Respon ini khusus untuk proxy-proxy HTTP
503 Service Unavailable	Server tidak mampu merepon request karena request sedang meluap.

b. *Field Header* merupakan informasi tambahan mengenai respon tersebut.
c. Data merupakan isi atau badan (body) dari respon

2.2.11 *Black Box Testing*

Black box testing berfokus pada pada kebutuhan fungsional perangkat lunak. Pada *black box testing*, memungkinkan pengembang perangkat lunak untuk mendapatkan *set* kondisi masukan yang sepenuhnya akan melaksanakan semua persyaratan fungsional suatu program. Metode ini memerlukan batas bawah dan batas atas dari data yang diharapkan, estimasi banyaknya data uji dapat dihitung melalui banyaknya *field* data entri yang akan diuji. Dengan metode ini dapat diketahui, jika fungsionalitas masih dapat menerima masukan data yang tidak diharapkan menyebabkan data yang disimpan kurang valid.

2.2.12 *White Box Testing*

White box testing juga dikenal sebagai pengujian struktural, pengujian *transparent box,* pengujian berbasis logika atau pengujian berbasis kode. Kata *white box* (kotak putih transparan) mengacu pada sebuah metode *test case*, perangkat lunak yang sedang diuji dianggap sebagai kotak (*box*). Sedangkan kata *white/transparent* mengacu pada bahwa kotak itu terlihat jelas isinya. Penguji dapat melihat jelas bagian dalam kotak dan cara kerja didalamnya. Seringkali *logical error* dan pemahaman *script* (kode program) yang keliru, akan berbanding terbalik dengan alur sistem yang seharusnya. Karena banyaknya yang mengatakan bahwa *logical path* adalah hal yang sulit untuk dieksekusi, sebenarnya dapat saja dieksekusi dalam keadaan normal. Kesalahan ketik dalam kode itu bersifat acak dan tidak dapat dihindari, dari banyak hal itulah maka kita harus melakukan

pengujian *white box*. Tujuannya adalah untuk memeriksa *logical path* perangkat lunak dengan memeriksa struktut logis perangkat lunak[22].

2.2.13 *System Usability Scale*

System usability scale dalam menentukan responden tidak memiliki konsep yang baku atau tidak ada penentuan secara khusus dari teori dasarnya. Kondisi tersebut disebabkan responden dari *system usability scale* merupakan pengguna akhir dari sebuah produk perangkat lunak yang akan dilakukan evaluasi atau pengujian. Dalam beberapa kajian, menunjukkan *system usability scale* menggunakan responden yang berbeda bahkan sampai dengan empat ratus sembilan puluh sembilan responden. Dalam pengujian yang lain, menggunakan responden sangat sedikit yaitu lima dan sepuluh responden. Penggunaan jumlah responden yang berbeda merupakan independensi dan sesuai dengan kebutuhan peneliti itu sendiri. Seorang peneliti diberikan kebebasan dalam menentukan responden sesuai dengan rumusan atau teknik pengambilan sampel penelitian[23].