PERANCANGAN APLIKASI DETEKSI KERENTANAN *SQL INJECTION* DAN *CROSS-SITE SCRIPT* PADA APLIKASI BERBASIS *WEBSITE* MENGGUNAKAN METODE *WATERFALL*

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh kelulusan Jenjang Strata Satu (S1)
Pada program Studi Teknik Informatika

Oleh :
ZAID MUSTOFA
361841001

SEKOLAH TINGGI MANAJEMEN INFORMATIKA & KOMPUTER
INDONESIA MANDIRI

2021

LEMBAR PENGESAHAN

PERANCANGAN APLIKASI DETEKSI KERENTANAN *SQL INJECTION* DAN *CROSS-SITE SCRIPT* PADA APLIKASI BERBASIS *WEBSITE* MENGGUNAKAN METODE *WATERFALL*

Oleh :

ZAID MUSTOFA
361841001

Tugas akhir ini telah diterima dan disahkan untuk memenuhi persyaratan mencapai gelar

SARJANA TEKNIK INFORMATIKA

Pada

PROGRAM STUDI TEKNIK INFORMATIKA SEKOLAH TINGGI MANAJEMEN INFORMATIKA & KOMPUTER INDONESIA MANDIRI

Bandung, Februari 2021
Disetujui Oleh

Ketua Program Studi,	Dosen pembimbing,
Chalifa Chazar S.T., M.T	Patah Herwanto S.T., M.Kom
NIDN : 0421098704	NIDN : 0027107501

LEMBAR PERSETUJUAN REVISI

PERANCANGAN APLIKASI DETEKSI KERENTANAN *SQL INJECTION* DAN *CROSS-SITE SCRIPT* PADA APLIKASI BERBASIS *WEBSITE* MENGGUNAKAN METODE *WATERFALL*

Oleh :

ZAID MUSTOFA
361841001

Telah melakukan sidang skripsi dan telah melakukan revisi sesuai dengan perubahan dan perbaikan yang diminta pada saat sidang skripsi

Bandung, Februari 2021
Menyetujui

No	Nama Dosen	Keterangan	Tanda Tangan
1	Patah Herwanto., S.T., M.kom	pembimbing	
2	Yudhi W. Arthana. R., S.T., M.Kom	Penguji 1	
3	Dr. Pahlawan Sagala	Penguji 2	

Bandung, Februari 2021
Disetujui Oleh

Ketua Program Studi,

Chalifa Chazar S.T., M.T
NIDN : 0421098704

SURAT PERNYATAAN

Dengan ini saya menyatakan bahwa :

(1) Naskah Skripsi ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik, baik di Sekolah Tinggi Manajemen Informatika dan Komputer Indonesia Mandiri maupun perguruan tinggi lainnya

(2) Skripsi ini murni merupakan karya penelitian saya sendiri dan tidak menjiplak karya pihak lain. Dalam hal ada bantuan atau arahan dari pihak lain maka telah saya sebutkan identitas dan jenis bantuannya di dalam lembar ucapan terima kasih

(3) Seandainya ada karya pihak lain yang ternyata memiliki kemiripan dengan karya saya ini, maka hal ini adalah diluar pengetahuan saya dan terjadi tanpa kesengajaan dari pihak saya

Pernyataan ini saya buat dengan sesungguhnya dan apabila di kemudian hari terbukti adanya kebohongan dalam pernyataan ini, maka saya bersedia menerima sanksi akademik sesuai norma yang berlaku di Sekolah Tinggi Manajemen Informatika dan Komputer Indonesia Mandiri.

Bandung, Januari 2021
Yang membuat Pernyataan

Zaid Mustofa
361841001

ABSTRAK

PERANCANGAN APLIKASI DETEKSI KERENTANAN *SQL INJECTION* DAN *CROSS-SITE SCRIPT* PADA APLIKASI BERBASIS *WEBSITE* MENGGUNAKAN METODE *WATERFALL*

Oleh

Zaid Mustofa
NIM 361841001

Tingginya akses terhadap *website* ini diiringi juga dengan tingginya pula tingkat keamanan nya. Menguji sistem keamanan sangat penting dari sekian faktor penyebab kurangnya keamanan *website* salah satunya adalah kesalahan penulisan kode program. Berbagai jenis serangan biasanya digunakan untuk menemukan celah keamanan seperti *SQL Injection* dan *Cross-site Script*.

Perancangan aplikasi menggunakan pendekatan *waterfall* dimulai dengan mengalisis masalah pada penelitian sebelumnya yaitu terdapat kekurangan pada metode *crawling* sehingga proses pencarian kerentanan menjadi kurang efektif. Dengan demikain membuat penulis tertarik dalam melakukan penelitian untuk merancang aplikasi untuk mendeteksi kerentanan pada jenis serangan *SQL injection* dan *Cross-Site Script* dan menganalisis kebutuhan untuk perancangan aplikasi yang akan dibangun.

Aplikasi yang dirancang akan memudahkan bagi para pengembang *website* sehingga dapat menekan terjadinya serangan *SQL injection* dan *Cross-Site Script*. Pada tahap pengujian aplikasi berhasil mendeteksi kerentanan terhadap *SQL Injection* dan *Cross-Site Script* pada *website* yang telah ditentukan sebelumnya.

Kata kunci : *Website*, Kerentanan, Keamanan. *Sql Injection, Cross-Site Script*

ABSTRACT

DESIGNING A SQL INJECTION VULNERABILITY DETECTION APPLICATION AND CROSS-SITE SCRIPT ON A WEBSITE-BASED APPLICATION USING WATERFALL METHOD

By

Zaid Mustofa
NIM 361841001

High access to this website is also accompanied by a high level of security. Testing the security system is very important. Of the many factors that cause the lack of website security. One of which is the error in writing program code. Various types of attacks are commonly used to find vulnerabilities such as SQL Injection and Cross-site Script.

Designing applications using the waterfall approach begins with analyzing the problem in previous research, namely that there are deficiencies in the crawling method so that the vulnerability search process becomes less effective. Thus making the author interested in conducting research to design applications to detect vulnerabilities in the type of SQL injection and Cross-Site Script attacks and analyze the need for designing applications to be built.

This application is designed to make it easier for website developers so that they can suppress SQL injection and Cross-Site Script attacks. At the testing stage the application successfully detected vulnerabilities to SQL Injection and Cross-Site Script on predetermined websites.

Keyword : *vulnerability, security, website, Sql Injection, Cross-Site Script*

KATA PENGANTAR

Dengan mengucapkan Alhamdulillah sebagai wujud syukur kepada Allah SWT, yang senantiasa memberikan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan laporan penelitian tugas akhir ini dengan baik dan tepat watku

Tugas akhir ini, berjudul PERANCANGAN APLIKASI DETEKSI KERENTANAN *SQL INJECTION* DAN *CROSS-SITE SCRIPT* PADA APLIKASI BERBASIS WEBSITE MENGGUNAKAN METODE *WATERFALL,* disusun untuk melengkapi tahapan akhir studi yang dijalani di Perguruan Tinggi Mnajemen Informatika dan Komputer Indonesia Mandiri

Tugas akhir ini berisi mengenai pengujian keamanan *website* pada jenis kerentanan *SQL injection* dan *Cross-Site Script* dengan harapan dapat memudahkan dalam pengecekan keamanan pada aplikasi berbasis web

Dengan segala keterbatasan tentunya diharapkan tugas akhir ini dapat bermanfaat bagi berbagai pihak, khususnya bagi penulis sendiri

Bandung, Januari 2021

Penulis

Zaid Mustofa
361841001

UCAPAN TERIMA KASIH

Puji dan syukur penulis panjatkan kepada Allah SWT. Karena atas limpahan rahmat dan karunia-Nya tugas akhir ini dapat terselesaikan dengan baik. Tak lupa penulis mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Bapak Prof. Dr. Chairuddin selaku ketua STMIK Indonesia Mandiri.
2. Ibu, Chalifa Chazar S.T., M.T. selaku Ketua Program Studi Teknik Informatika
3. Bapak, Patah Herwanto S.T. M.Kom Selaku dosen pembimbing yang telah banyak membantu dan memberikan bimbingan selama pembuatan tugas akhir ini.
4. Seluruh dosen, Staff dan karyawan STMIK Indonesia Mandiri yang telah mendidik dan membantu dalam memberikan informasi serta motivasi dalam proses studi maupun tugas akhir berlangsung
5. Orang tua dan istri fitriyani . yang senantiasa mendoakan dan memberikan bantuan moril maupun materil serta putri tersayang diaz zahida dan izma fitri hazimah.
6. Kang Budi S.T., F. Bagas Samudra S.T, dan teman-teman yang telah memberikan dukungan serta memotivasi penulis untuk menyelesaikan tugas akhir.
7. Dan terakhir kepada semua pihak yang tidak bisa penulis sebutkan satu persatu atas semua bantuannya penulis mengucapkan banyak terima kasih.

Semoga Allah SWT memberikan balasan yang berlipat ganda kepada semuanya. Demi perbaikan selanjutnya, saran dan kritik yang membangun akan penulis terima dengan senang hati. Akhirnya, hanya kepada Allah SWT penulis serahkan segalanya, semoga dapat bermanfaat khususnya bagi penulis umumnya bagi kita semu

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I

PENDAHULUAN

1.1.Latar Belakang

Di era new normal sekarang ini, semua orang bergantung pada internet. kehidupan berubah menjadi serba online. Sebagai contoh di sektor Pendidikan metode pembelajaran dialihkan menjadi secara online. Tingginya akses terhadap *website* ini diiringi juga dengan tingginya pula tingkat keamanan nya.

Berdasarkan informasi dari Pusat Opersi Keamanan Siber Nasional (Pusopskamsinas) dan Badan Siber Sandi Negara (BSSN) Tercatat 88.414.296 serangan siber yang terjadi dari 1 Januari hingga 12 April 2020. Terpantau 25.224.811 serangan di bulan Januari, kemudian tercatat 29.188.645 serangan di bulan Februari, lalu 26.423.989 serangan terjadi di bulan Maret hingga 12 April 2020. 7.576.851 serangan. (BSSN. 2020)

Menguji sistem keamanan sangat penting dari sekian faktor penyebab kurangnya keamanan website salah satunya adalah kesalahan penulisan kode program. Kesalahan saat menulis kode program dalam pembuatan *website* adalah sesuatu yang sering digunakan penyerang, yang mengarah pada penggunaan kesalahan ini untuk menyerang *website* tersebut. Berbagai jenis serangan biasanya digunakan untuk menemukan celah keamanan. Jenis serangan ini termasuk *SQL Injection, Cross-Site Script*. Berdasarkan *report* OWASP 2003 sampai 2017 tentang 10 risiko keamanan aplikasi bersis web paling kritis adalah serangan *SQL Injection* dan *Cross-Site Script* berada di urutan teratas.

Oleh karena itu untuk menjaga keamanan *website*, perlu dilakukan pengujian aplikasi secara teratur. Ini penting karena jika tidak ada pengujian rutin, tidak ada jaminan bahwa situs tersebut akan aman dan terlindungi dari serangan dalam jangka panjang.

(Elu, A, M. 2013) Melakukan penelitian tentang perancangan aplikasi untuk mendeteksi kerentanan *SQL Injection* yang bernama *SVSQL Injection.* Aplikasi ini

digunakan untuk mendeteksi keamanan *website* dari serangan *SQL Injection*. Aplikasi yang dirancang hanya menggunakan satu jenis serangan yaitu *SQL Injection.*

(Yudha, F. Panji, A, M. 2018) juga melakukan penelitian tentang perancangan aplikasi untuk menguji keamanan *website* terhadap kerentanan *Phising Cross-Site Scripting, SQL Injection,,* aplikasi yang dirancang menggunakan Bahasa *python*. Terdapat kekurangan pada teknik *crawling* hanya bisa menelusuri halaman depan dan metode *SQL Injection* dan *Cross-Site Script* pada *website* yang terdapat *page* "*id*" saja sedangkan ketika *website* tidak memiliki *page "id"* nya di anggap tidak rentan sehingga kurang efektif untuk mendeteksi kerentanan.

Oleh karena itu dengan dilakukan nya penyusunan skripsi ini untuk melakukan perencanaan aplikasi untuk mendeteksi kerentanan pada *website* terhadap serangan *SQL Injection, Cross-Site Scripting (XSS),* dan melengkapi pada penelitian sebelumnya.

1.2. Identifikasi Masalah

1. Kurangnya kesadaran pengembang aplikasi terhadap keamanan pada *website* yang dibangunnya
2. Masih tingginya serangan pada kerentanan *website* pada jenis serangan *SQL Injection* dan *Cross-Site Script*

1.3 Tujuan Penelitian

1. Merancang sebuah aplikasi yang memudahkan dalam pengujian keamanan *website*
2. Meminimalisir terjadinya serangan *SQl Injection dan Cross-Site Script.*

1.4. Batasan Masalah

Supaya penelitian ini tidak melebar terlalu jauh maka Batasan masalah dibatasi pada :

1. Hanya menguji pada aplikasi berbasis *website* yang memiliki *form* input *HTML*.

2. Metode pengujian keamanan *SQL Injection, Cross-Site Script (XSS).*
3. Aplikasi sebatas mendeteksi kerentanan dan tidak melakukan *exploitasi* secara langsung
4. Dirancang untuk para pengembang aplikasi *website*

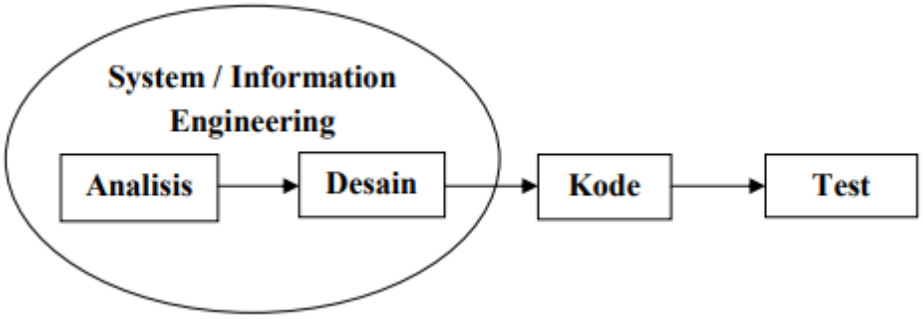
1.5 Metode Penelitian

1. Studi Pustaka

Menurut Moh. Nazir Studi pustaka yaitu pengumpulan data, melalui studi terhadap buku, dokumen, catatan dan laporan yang berkaitan dengan masalah penelitian.

2. Metode Pengembangan Perangkat Lunak

Pengembangan aplikasi yang dirancang menggunakan pendekatan *waterfall*. Metode *waterfall* adalah metode pengembangan perangkat lunak parsial yang sistematis, mulai dari tingkat dan kemajuan sistem hingga analisis, desain, kode, pengujian, dan pemeliharaan. Berikut tahapan pembuatan model air terjun (A.S Rosa, Shalahuddin, M. 2018 : 28).



Gambar 1.1 Model *Waterfall*

1. Analisis

   untuk merencanakan kebutuhan sistem yang akan dibangun sehingga sistem yang sesuai dapat dikembangkan.
2. Desain

   Tahap perancangan meliputi perancangan antarmuka dan rancangan komponen antarmuka, serta alur program perancangan

3. Kode

   Pada tahap ini, hasil desain di implementasikan sebagai baris kode program yang dapat dipahami oleh komputer.

4. Test

   Tahap pengujian dilakukan untuk menguji semua fungsi dari aplikasi yang telah di rancang

1.6 Sistematika penulisan

Penulisan sistematis penelitian ini bertujuan untuk memberikan gambaran tentang penelitian yang dilakukan, seperti gambar dibawah ini :

BAB I PENDAHULUAN

Bab ini memperkenalkan latar belakang masalah, identifikasi masalah, penyajian masalah, batasan masalah, tujuan penelitian, metode penelitian dan sistem penulisan.

BAB II LANDASAN TEORI

Bab ini akan menjelaskan tentang landasan teori yang berkaitan tentang pembahasan dalam penelitian ini seperti tentang keamanan *website*, teori dasar *SQL Injection, Cross-Site Scripting (XSS)*

BAB III ANALISA MASALAH DAN PERANCANGAN PROGRAM

Bab ini menjelaskan analisis aplikasi yang berjalan dan aplikasi yang akan dirancang

BAB IV IMPLEMENTASI DAN UJI COBA

Bab ini berisi tentang proses implementasi dari aplikasi yang telah dirancang sebelumnya

BAB V PENUTUP

Bab ini berisi tentang kesimpulan, saran dalam uraian pembahasan penelitian, agar hasil penelitian selanjutnya menjadi lebih baik.

BAB II

LANDASAN TEORI

2.1 Penelitian Terkait

(Elu A, M. 2013) melakukan sebuah penelitian perancangan aplikasi bernama *SVSQL Injection* untuk mentedeksi kerentanan *SQL Injection* untuk kemanan *website.* Aplikasi akan menguji pada pengujian komentar baris, pengujian perintah tumpukan, pengujian kalimat, pengujian integer, pengujian string, pengujian penggabungan kueri, dan pengujian error.

(Yudha, F. Panji A, M. 2018) juga melakukan penelitian tentang perancangan aplikasi untuk mendeteksi kerentanan *website* , aplikasi yang di rancang untuk mendeteksi kerentanan *Cross-Site Scripting ,Phising , SQL Injection, , dan*. Aplikasi dibangun menggunakan Bahasa *Python* di dalamnya terdapat kolom masukan url target dan tombol *attack.*

Beberapa Referensi ada pada table berikut ini :

Table 2.1 Tabel Referensi

No	Penulis	Judul	Pembahasan
1	Sholeh, A, N. Wardaya S, S, M. (2019)	Analisis dan pengujian kerentanan sistem informasi perpustakaan	Pemindaian celah kemanan menggunakan *tools vulnerability scanner*
2	Elu, A, M. (2013)	Rancang bangun aplikasi pendeteksian vulnerability structured query language (SQL) Injection untuk keamanan website	Merancang sebuah aplikasi bernama SVSQL untuk mengecek keamanan website dari kerentanan SQL Injection
4	Yudha, F. Panji A, M. (2018)	Perancangan aplikasi pengujian celah keamanan pada aplikasi berbasis web	Merancang aplikasi vulnerability scan menggunakan Bahasa python
5	Dahlan, M. Latubessy,	Pengujian Dan Analisa Keamanan Website	Pemantauan serangan menggunakan tools IDS

	A. Nurkamid, M. Anggraini H, L. (2014)	Terhadap Serangan SQL Injection (Studi Kasus : Website UMK)	Snort, sehingga bisa mengetahui serangan apa saja yang mencoba masuk kedalam sistem
6	Kusrini (2019)	Mendeteksi kerentanan keamanan aplikasi website menggunakan metode owasp	Diperlukan penilaian risiko kerentanan keamanan untuk aplikasi berbasis situs web sehingga sebelum mengunggah aplikasi berbasis situs web ke server produksi, potensi risiko keamanan dapat dilihat untuk mencegah dan menyelesaikan risiko keamanan
7	Rodriguez, G. Torres, G, J. Flores, F. Benavides, E. (2019)	*Cross-Site Scripting (XSS) Attacks And Mitigation: A Survey*	Survey tentang serangan *Cross-Site Script*
8	Justin Clarke (2009)	*SQL Injection Attack and Defense*	Membahas tentang cara kerja sql injection dan cara menamngkal nya
9	Seth Fogie (2007)	*XSS Attack*	Membahas tentang cara kerja *Cross-Site Script*

2.2 Pengertian *Website*

Menurut Gregorius (2000) website adalah kumpulan halaman web yang saling berhubungan. Web terdiri dari satu atau lebih halaman dan berisi sekumpulan halaman yang disebut home page. Halaman beranda ada di bagian atas dan halaman terkait ada di bagian bawah. Biasanya, setiap halaman di bawah halaman utama disebut subhalaman, yang berisi hyperlink ke halaman lain di Internet

2.3 Keamanan Informasi

Menurut G. J. Simons, keamanan informasi mengacu pada pekerjaan mencegah penipuan (fraud) atau mendeteksi penipuan dalam sistem berbasis informasi di mana informasi itu sendiri tidak memiliki arti fisik.

Aspek-aspek yang harus dipenuhi oleh sistem untuk menjamin keamanan informasi antara lain:

informasi yang diberikan akurat dan lengkap (informasi yang benar), informasi yang dimiliki oleh orang yang tepat, dan informasi yang dapat diakses dan digunakan sesuai kebutuhan (pada waktu yang tepat). , Dan format informasi yang benar (format yang benar). Saat membuat program keamanan informasi, beberapa prinsip dasar harus dipenuhi agar sistem dapat diandalkan.

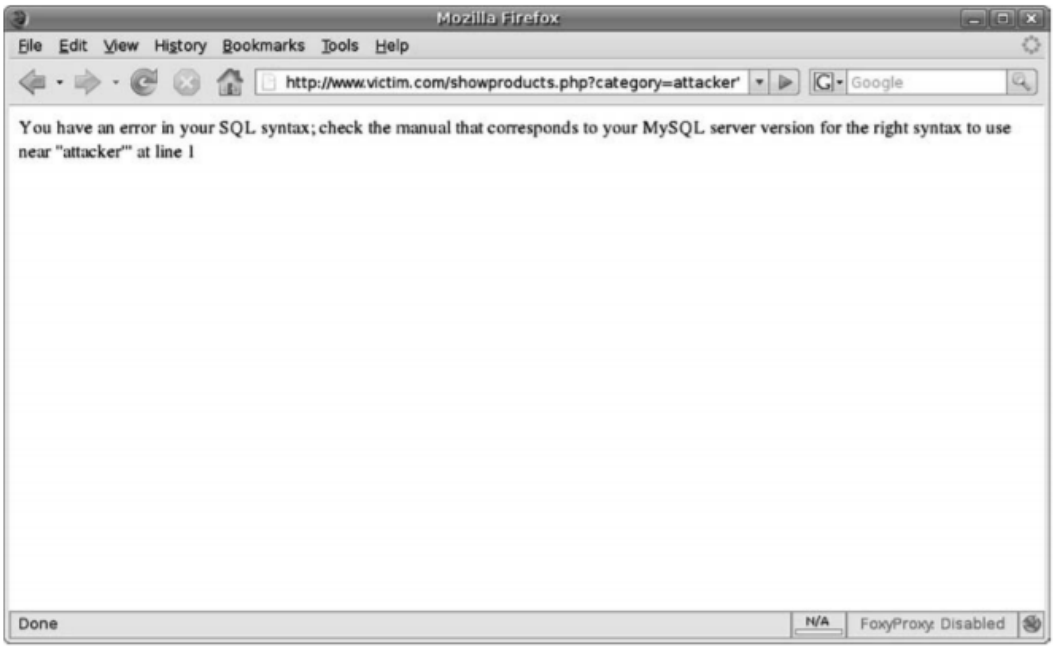
2.4 Kerentanan (*Vulnerability)*

Menurut Veracode Kerentanan aplikasi adalah kelemahan atau kelemahan sistem dalam aplikasi yang dapat dieksploitasi untuk membahayakan keamanan aplikasi. Setelah penyerang menemukan cacat, atau kerentanan aplikasi, dan menentukan cara mengaksesnya, penyerang berpotensi mengeksploitasi kerentanan aplikasi untuk memfasilitasi kejahatan dunia maya. Kejahatan ini menargetkan kerahasiaan, integritas, atau ketersediaan (dikenal sebagai "triad CIA") sumber daya yang dimiliki oleh aplikasi, penciptanya, dan penggunanya. Penyerang biasanya mengandalkan alat atau metode tertentu untuk melakukan penemuan dan penyusupan kerentanan aplikasi.

2.4.1 *SQL Injection*

Menurut Justin Clarke (2009) *SQL Injection* adalah teknik serangan yang mengeksploitasi kode dengan memodifikasi backend SQL dengan Memasukkan kalimat manipulasi.

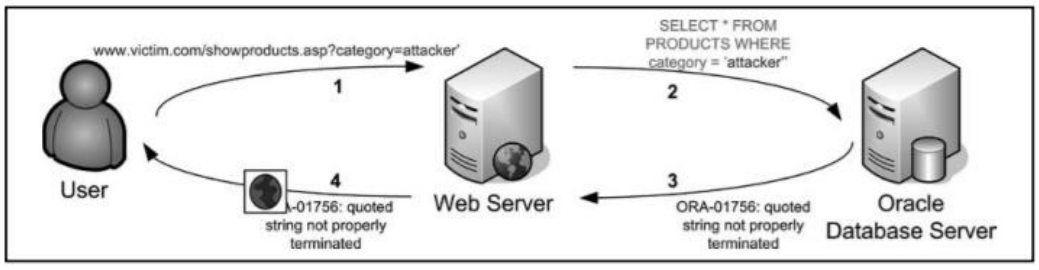
Dengan mencoba manipulasi parameter pada url target *http://www.victim.com/showproducts.php?category=attacker'* dengan menginputkan tanda petik satu (‘) pada akhiran url atau pada *form* jika *website* target menampikan peringantan kesalahan database MySQL seperti pada gambar 2.1, maka website tersebut rentan *SQL Injection* :



Gambar 2.1 Contoh *MySQL error* (Justin Clarke)

2.4.1.1 Alur *SQL Injection*

SQL Injection memungkinkan peretas untuk masuk kedalam sistem database tanpa melalui authentikasi dan bisa menghapus atau mengunduh semua data yang tersimpan di database. Jika hal ini terjadi dan tidak ada backup database, akan sangat berbahaya. Untuk tujuan keamanan data, perlu melakukan backup data pada *storage external* atau *cloud.*



Gambar 2.2 Alur *SQL Injection* (Justin Clarke. 2009)

Alur ketika *website* merespon *SQL injection error* :

1. Pengguna mengirimkan permintaan dalam upaya untuk mengidentifikasi kerentanan injeksi SQL. Dalam kasus ini, pengguna mengirimkan nilai dengan satu kutipan yang ditambahkan padanya
2. Server Web mengambil data pengguna dan mengirimkan kueri SQL ke server database. Dalam contoh ini, bahwa pernyataan SQL yang dibuat oleh server Web termasuk input pengguna dan membentuk kueri yang salah secara sintaksis karena keduanya mengakhiri kutipan
3. Server database menerima kueri SQL dalam format yang salah dan mengembalikan kesalahan ke Server web
4. Server Web menerima kesalahan dari database dan mengirimkan respons HTML kepada pengguna. Dalam hal ini, itu mengirim pesan kesalahan, tetapi sepenuhnya terserah aplikasi bagaimana menampilkan kesalahan apa pun dalam konten respons HTML (justin Clarke. 2009)

2.4.2 *Cross-Site Scripting (XSS)*

XSS adalah teknik serangan yang memaksa situs Web untuk menampilkan kode berbahaya, yang kemudian dijalankan di browser Web pengguna. Pertimbangkan bahwa kode eksploitasi XSS, biasanya (tetapi tidak selalu) ditulis dalam Hypertext Markup Language (HTML) / JavaScript (alias JavaScript malware software [malware]), tidak dijalankan di server. Server hanyalah host, sedangkan serangan dijalankan dalam browser Web. Hacker hanya menggunakan situs Web terpercaya sebagai saluran untuk melakukan serangan. Pengguna adalah korban yang dituju, bukan server. Sekali penyerang memiliki rangkaian kontrol di browser Web pengguna, mereka dapat melakukan banyak tindakan jahat termasuk pembajakan akun, perekaman penekanan tombol, peretasan intranet, riwayat pencurian, dan sebagainya. Bagian ini menjelaskan berbagai cara yang mungkin dilakukan pengguna menjadi XSS dan mengontrak muatan malware JavaScript (Seth Fogie. 2007)

2.4.2.1 *Reflected XSS*

Menurut (G Rodriguez, J Torres, E Benavides. 2019) jenis ini, penyerang menempatkan skrip untuk mencuri cookie korban, untuk mempersonifikasikan dirinya sendiri seolah-olah itu adalah sesinya. Dengan cookie yang diterima, penyerang dapat melakukan tindakan menggunakan izin korban tanpa menggunakan jenis kata sandi. Serangan ini biasa terjadi di mesin pencari, biasanya, kode disuntikkan melalui formulir, URL, cookie, program atau bahkan video. Serangan ini mengeksploitasi kerentanan di aplikasi Web yang menggunakan (atau mencerminkan) informasi yang disediakan oleh pengguna untuk menghasilkan halaman keluar. Dengan cara ini, kode dialihkan melalui mekanisme ketiga. Misalnya melalui spoofing (e-mail). Dengan ini, penyerang dapat meyakinkan pengguna untuk mengklik link di pesan untuk menjalankan kode JavaScript apa pun. Konsekuensinya adalah pengalihan lalu lintas pengguna ke aplikasi web penyerang. Jika aplikasi Web tersebut menunjukkan kerentanan XSS, pelaksanaannya akan dilakukan dalam lingkungan tepercaya dari situs Web yang menghosting aplikasi tersebut.

Sebagai contoh Pada gambar 2.3 menggambarkan apa yang terjadi ketika pada form pencarian di sisipkan script XSS sebagai contoh “><SCRIPT>alert(‘XSS%testing’)</SCRIPT>. Halaman web merespon kode script tadi dengan menampilkan JavaScript yang telah di sisipkan sebelumnya.

Gambar 2.3 Contoh serangan XSS (Seth Fogie. 2007)

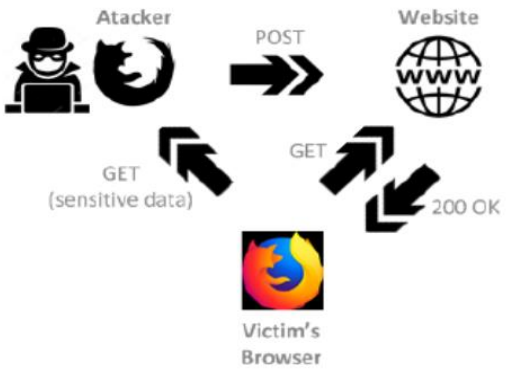
2.4.2.2 Stored XSS

Menurut (G Rodriguez, J Torres, E Benavides. 2019) Penyerang memasukkan kode HTML berbahaya langsung ke halaman web atau situs yang mengizinkannya (situs rentan). Dalam serangan ini membutuhkan tag pemrograman (script seperti JavaScript). Kode-kode ini dibuat permanen di web untuk semua pengguna setelah menjalankan serangan pertama. Akibatnya, setiap kali seseorang memasuki bagian di mana ada kode yang diinjeksi, ini akan dijalankan di browser mereka dan mereka akan mematuhi tindakan yang diprogram dalam skrip mereka. Varian ini lebih berbahaya karena didasarkan pada injeksi kode berbahaya di konten yang disimpan di server aplikasi web eksternal. Artinya, data yang dikirim oleh penyerang disimpan secara permanen di server dan kemudian ditampilkan kepada pengguna yang mengunjungi website tersebut. Diantara konsekuensinya adalah: memungkinkan eksekusi kode untuk mendapatkan atau meningkatkan hak istimewa yang lebih tinggi. Pengguna default telah mengaktifkan akun administrator mereka. Untuk ini, selalu disarankan untuk

menonaktifkan eksekusi JavaScript dari browser, namun analisis yang lebih dalam diperlukan karena kebutuhan interaksi dengan situs web.

2.4.2.3 DOM XSS

Menurut (G Rodriguez, J Torres, E Benavides. 2019) Dikenal sebagai Type 0 atau DOM-Based XSS, ini dianggap serangan yang lebih rumit dan sedikit diketahui atau umum. Perbedaannya adalah bahwa kode berbahaya disuntikkan melalui URL tetapi tidak dimuat sebagai bagian dari web dalam kode sumbernya. Deteksinya lebih sulit karena maliciousload tidak mencapai server. Itu dianggap sebagai XSS lokal karena kerusakan disebabkan oleh skrip yang ada di sisi klien. Pada dasarnya ketika halaman yang terinfeksi dibuka, kode berbahaya mengeksploitasi beberapa kerentanan untuk menginstal dirinya sendiri di file



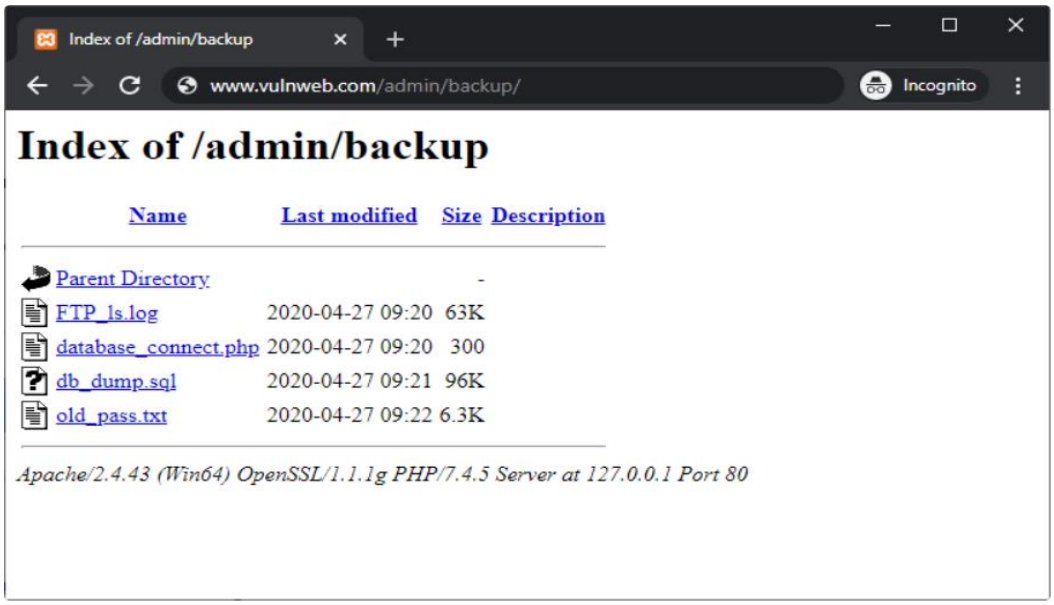
Gambar 2.4 Sekenario serangan XSS (G Rodriguez, J Torres, E Benavides. 2019)

2.4.3 *Directory Listing Enable*

Menurut (acunetix. 2020) Serangan ini menampilkan isi direktori ketika tidak ada file indeks di direktori situs web tertentu. Bahaya membiarkan fungsi ini diaktifkan adalah bisa menungkapkan informasi yang ada pada server web tersebut. Misalnya, Ketika pengguna meminta akses ke situs www.acunetix.com tanpa menentukan file indeks seperti index.php, index.html, default.asp, server web memproses permintaan ini, mengembalikan file indeks untuk direktori itu, dan

browser menampilkan situs web. Namun, jika file indesk tidak ada dan jika daftar direktori diaktifkan, server web akan mengembalikan konten direktori sebagai gantinya .

Sebagai contoh seorang pengguna membuat permintaan situs web ke www.vulnweb.com/admin/ . respin dari server menyertakan konten direktori daridirektori, seperti yang terlihat pada gambar 2.5 :



Gambar 2.5 Contoh *directory listing enable* (acunetix, 2020)

2.4.4 *Insecure Direct Object Reference*

Menurut (OWASP. 2020). *Insecure Direct Object References* Terjadi saat pengembang memaparkan referensi ke objek Implementasi internal, seperti file, direktori, atau kunci database. Tidak dicentang Kontrol akses atau perlindungan lainnya, penyerang dapat memanipulasi referensi ini Akses data yang tidak sah

Dan memungkinkan penyerang melewati otorisasi dan mengakses sumber daya secara langsung dengan mengubah nilai parameter yang digunakan untuk mengarahkan langsung ke objek. Sumber daya tersebut dapat berupa entri basis data pengguna lain, file dalam sistem, dan sebagainya. Ini karena aplikasi memasukkan

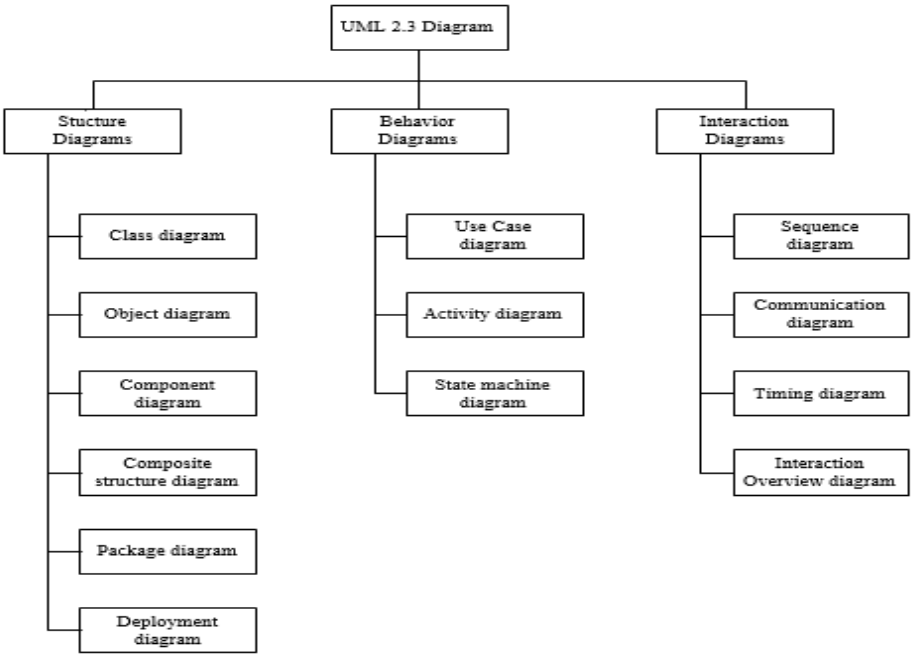
input yang diberikan oleh pengguna dan menggunakannya untuk mengambil objek tanpa melakukan pemeriksaan otorisasi yang memadai.

2.5 *Unified Modeling Language (UML)*

Menurut Rubaugh, J. Jacobson, I. Booch Grady (1998) dalam bukunya yang berjudul *"The Unified Modeling Language Reference Manual"* menjelasken bahwa Unified Modeling Language (UML) adalah bahasa pemodelan visual universal yang digunakan untuk mendefinisikan, memvisualisasikan, membuat, dan merekam artefak produk. Sistem perangkat lunak. Ini menangkap keputusan dan pemahaman tentang sistem mana yang harus digunakan Dibangun. Ini digunakan untuk memahami, merancang, mencari, mengkonfigurasi, memelihara dan Kontrol informasi tentang sistem. Ini dirancang untuk digunakan dengan semua metode pengembangan, tahapan siklus hidup, domain aplikasi dan media. pemodelan Bahasa yang dirancang untuk menyatukan pengalaman masa lalu dalam teknologi pemodelan, dan Integrasikan praktik terbaik perangkat lunak saat ini ke dalam metode standar. UML mencakup konsep, simbol, dan pedoman semantik. Ini memiliki bagian statis, dinamis, lingkungan dan organisasi. Didesain untuk didukung secara interaktif Alat pemodelan visual dengan encoder dan penulis laporan.

2.5.1 *Diagram UML*

Pada UML 2.3 berisi 13 grafik yang terbagi dalam 3 kategori. Kategori dan jenis grafik adalah sebagai berikut



Gambar 2.6 Diagram UML (Rosa, A.S., Shalahuddin, M. 2018 : 140)

Berikut penjelasan dari pembagian kategori tersebut.

1. *Structure diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan struktur statis dari sistem yang akan dimodelkan.
2. *Behavior diagrams* yaitu sekumpulan diagram yang digunakan untuk menggambarkan perilaku sistem atau rangkaian perubahan sistem.
3. *Interaction diagram* yaitu Sekumpulan diagram yang digunakan untuk menggambarkan interaksi antara sistem dan sistem lain serta interaksi antar subsistem dalam sistem.

2.5.2 *Class Diagram*

Class diagram menggambarkan struktur sistem dengan menentukan kelas-kelas yang akan digunakan untuk membangun sistem

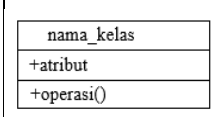
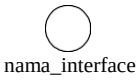
Kelas memiliki apa yang disebut properti, metode, atau operasi

1. Atribut adalah variabel yang dimiliki oleh kelas.
2. Operasi atau metode adalah fungsi kelas.

Berikut adalah symbol pada class diagram

Tabel 2.2 Simbol pada *Class diagram*

(Rosa, A.S., Shalahuddin, M., 2018 : 146)

Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem
antarmuka/ *interface* 	Mirip dengan konsep antarmuka dalam pemrograman berorientasi objek
asosiasi/ *association*	Hubungan antar kelas memiliki arti umum, dan hubungan ini biasanya disertai dengan keberagaman
asosiasi berarah/ *directed association* 	Hubungan antara kelas dengan satu jenis makna digunakan oleh kelas lain, dan asosiasi tersebut biasanya disertai dengan multiplisitas
Generalisasi 	Hubungan antar kelas dengan makna generalisasi-spesialisasi (generalisasi khusus)
Kebergantungan/ *Dependency* 	Hubungan antar kelas dan ketergantungan antar kelas
Agregasi/ *aggregation* 	Hubungan antar kelas yang memiliki makna sebagian (whole part)

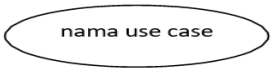
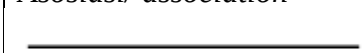
2.5.3 Use Case Diagram

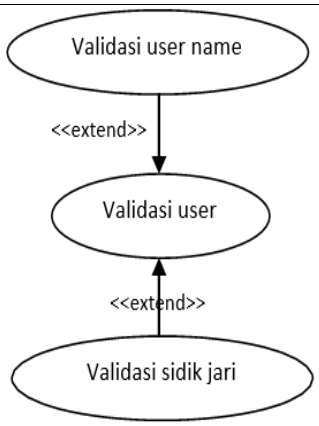
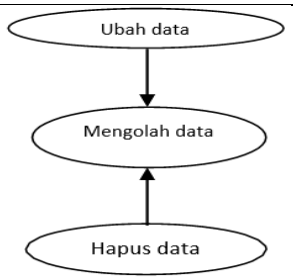
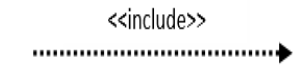
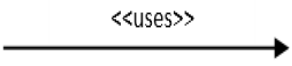
Use case menggambarkan interaksi antara satu atau lebih partisipan dan sistem informasi yang akan dibuat. Secara garis besar use case digunakan untuk menentukan fungsi-fungsi dalam suatu sistem informasi dan siapa yang berhak menggunakan fungsi tersebut.

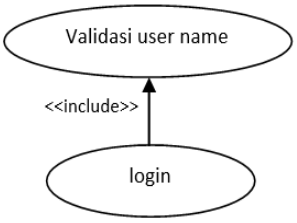
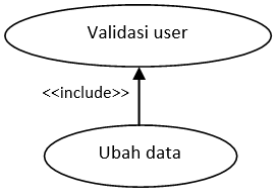
Berikut adalah simbol dalam use case diagram :

Tabel 2.3 Simbol pada *Use Case* Diagram

(Rosa, A.S., Shalahuddin, M., 2018 : 156)

Simbol	Deskripsi
Use Case 	Fungsi yang disediakan oleh sistem sebagai satu kesatuan dapat bertukar pesan antar unit atau partisipan; biasanya dengan Gunakan kata kerja di awal atau awal frasa dalam nama kasus penggunaan
Aktor/ actor  Nama aktor	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi akan dibuat di luar sistem informasi itu sendiri, oleh karena itu, meskipun simbol aktor adalah gambaran dari suatu karakter, aktor tersebut belum tentu karakter. Biasanya menggunakan kata ini Objeknya ada di awal frasa nama aktor.
Asosiasi/ association 	Komunikasi antara aktor yang berpartisipasi dalam use case atau use case dan use case yang dimiliki Interaksi dengan aktor
Ekstensi/ extend	Komunikasi dengan kasus penggunaan tambahan dari kasus penggunaan, di mana meskipun tidak ada kasus penggunaan tambahan, kasus penggunaan tambahan dapat ada secara independen; mirip dengan prinsip pewarisan dalam pemrograman berorientasi objek; biasanya, kasus penggunaan lain memiliki nama gudang perangkat lunak yang sama dengan kasus penggunaan tambahan, misalnya:

	 Panah menunjuk ke kasus penggunaan tambahan
Generalisasi/ generalization	Misalnya, hubungan antara generalisasi dan spesialisasi (umum-spesifik) antara dua kasus penggunaan, di mana satu fungsi lebih umum daripada yang lain.
	 Arah panah menunjuk ke use case, dan use case menjadi generalisasinya (umum).
Menggunakan/ include/ uses  	Hubungan dengan use case lain yang telah ditambahkan ke use case yang membutuhkan use case untuk menjalankan fungsinya atau sebagai syarat untuk menjalankan use case ini. Terkait kasus penggunaan, ada dua tampilan yang cukup besar: 1. Include berarti bahwa use case yang ditambahkan akan selalu dipanggil ketika use case tambahan dijalankan, misalnya dalam kasus berikut ini:

	 2. Include artinya use case tambahan akan selalu mengecek apakah use case tambahan sudah dieksekusi sebelum use case tambahan dieksekusi, misalnya pada case berikut:
	 Berdasarkan pertimbangan dan penjelasan yang diperlukan, kedua penjelasan di atas dapat menjadi salah satu dari dua

2.5.4 *Activity Diagram*

Menggambarkan aktivitas suatu sistem atau proses bisnis. Oleh karena itu, aktivitas yang dapat dilakukan sistem.

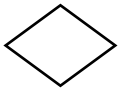
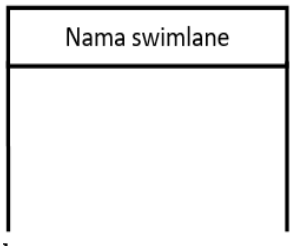
Activity Diagram juga dapat digunakan untuk mendefinisikan hal- hal berikut :

1. Desain proses bisnis, menjelaskan urutan proses bisnis sistem
2. Urutan tampilan sistem / antarmuka, dimana setiap aktivitas dianggap memiliki desain antarmuka tampilan.
3. Desain pengujian yang menganggap bahwa setiap aktivitas perlu diuji dan perlu didefinisikan sebagai kasus uji.

Berikut ini simbol pada *activity diagram*.

Tabel 2.4 Simbol pada *Activity Diagram*

(Rosa, A.S., Shalahuddin, M., 2018 : 162)

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem,
Aktivitas 	Aktivitas dilakukan oleh sistem, dan aktivitas biasanya dimulai dengan kata kerja.
Percabangan/ *decision* 	Asosiasi cabang, jika ada beberapa opsi aktif.
Penggabungan/ *join* 	Asosiasi yang menggabungkan lebih dari satu aktivitas menjadi satu.
Status akhir 	Keadaan akhir sistem, diagram aktivitas memiliki keadaan akhir.
Swimlane  atau 	Memisahkan organisasi bisnis yang bertanggung jawab atas aktivitas yang terjadi.

2.5.5 Sequence Diagram

Menggambarkan perilaku objek dalam use case dengan menjelaskan siklus hidup objek dan pesan yang dikirim dan diterima di antara objek. Oleh karena itu,

untuk menggambar diagram urutan, Anda harus mengetahui objek yang terlibat dalam kasus penggunaan dan metode yang termasuk dalam kelas yang digunakan oleh objek tersebut.

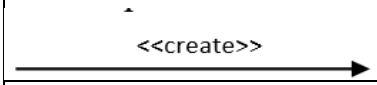
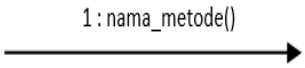
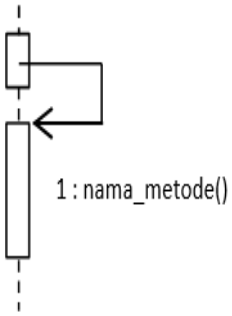
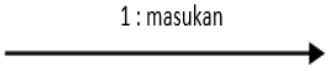
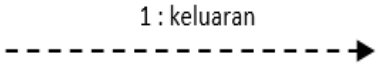
Jumlah sequence diagram yang harus digambar adalah sebanyak definisi use case dan alurnya masing-masing. Yang terpenting adalah semua use case telah diidentifikasi dan interaksi jalur pesan dimasukkan ke dalam sequence diagram. Oleh karena itu, semakin banyak use case teridentifikasi, semakin banyak pula operasi yang harus dilakukan..

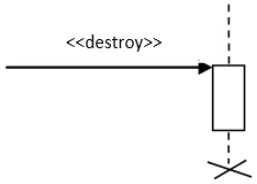
Berikut ini simbol pada sequence diagram.

Tabel 2.5 Simbol pada Diagram Sequence

(Rosa, A.S., Shalahuddin, M., 2018 : 165)

Simbol	Deskripsi
Aktor  nama aktor atau nama aktor tanpa waktu aktif	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi akan dibuat di luar sistem informasi itu sendiri. Oleh karena itu, meskipun lambang seorang aktor merupakan gambaran dari seorang tokoh, seorang aktor belum tentu merupakan seorang tokoh. Biasanya, kata benda digunakan di awal frasa nama aktor
Garis Hidup/ *lifeline* 	Merepresentasikan kehidupan objek tersebut.
Objek nama objek : nama kelas	Merepresentasikan objek yang berinteraksi dengan pesan

Waktu Aktif 	Deklarasikan objek aktif dan tukar pesan
Pesan tipe *create* 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.
Pesan tipe *call* 	Menyatakan objek Operasi / metode panggilan Di objek lain atau dirinya sendir  Arah / panah menunjuk ke objek melalui operasi / metode, karena menurut diagram kelas objek interaktif, operasi / metode dipanggil, dan operasi / metode yang dipanggil harus ada pada diagram kelas
Pesan tipe send, 	Menyatakan bahwa satu objek mengirimkan data / input / informasi ke objek lain, arah Panah menunjuk ke objek yang akan dikirim.
Pesan tipe return 	Ketika sebuah objek yang telah melakukan operasi atau metode mengembalikan objek tertentu, panah menunjuk ke objek yang menerima nilai pengembalian.

Pesan tipe destroy  	Nyatakan bahwa suatu objek mengakhiri umur objek lain, dan panah menunjuk ke objek yang diakhiri. Sebaiknya ada objek yang dibuat, lalu ada objek yang hancur.

2.6 MySQL

Menurut Budi Raharjo (2015) MySQL adalah RDBMS (database server), dapat mengatur database dengan sangat cepat, Bisa menampung banyak data dan bisa diakses banyak orang pengguna.

Menurut agus Saputra (2012) didefinisikan dalam bukunya MySQL Merupakan salah satu database kelas satu dunia, sangat cocok untuk penggunaan gabungan Gunakan bahasa pemrograman PHP.

Menurut definisi Heni A. Puspitosari (2011) MySQL Merupakan salah satu perangkat lunak yang paling banyak digunakan untuk server database, MySQL adalah open source dan menggunakan SQL.

2.7 *Hypertext Preprocessor (PHP)*

PHP adalah bahasa skrip slide server, bahasa pemrograman yang digunakan untuk mengembangkan situs web statis atau situs web dinamis atau aplikasi web. PHP adalah singkatan dari Hypertext Preprocessor, sebelumnya dikenal sebagai "beranda pribadi"

2.8 Apache

Apache adalah server web, sangat populer dan paling banyak digunakan di dunia. Dikelola oleh Apache Software Foundation, Apache diluncurkan pada tahun 1995 dan tetap populer selama setahun setelah itu hingga hari ini. Sebagai fungsi web browser, Apache bertindak sebagai penghubung antara pengguna (browser) dan server. Apache awalnya dikembangkan sebagai server web open source untuk sistem operasi seperti UNIX. Apache telah banyak digunakan oleh banyak perusahaan besar seperti Linkedin, Adobe, General Electric, IBM, dll., Dan

penyedia layanan panel kontrol juga menggunakan Apache sebagai browser web mereka.

2.9 Pengujian Perangkat Lunak

2.9.1 Definisi

Definisi menurut (perry, W. E. 1990) Pengujian perangkat lunak adalah proses menjalankan program atau sistem yang bertujuan untuk menemukan atau melibatkan setiap aktivitas yang bertujuan untuk mengevaluasi sifat atau fungsi program atau sistem dan menentukan hasil yang memenuhi kebutuhan perusahaan..

2.9.2 Tujuan Pengujian aplikasi

1. Menemukan beberapa kemungkinan kesalahan dalam perangkat lunak yang diracang.
2. Menguji fungsi dari aplikasi tersebut

2.9.3 *White Box Testing*

White Box Testing Ini adalah metode pengujian yang digunakan untuk memeriksa kode program yang ada dan menganalisis kesalahan dengan melihat modul. Jika output yang dihasilkan oleh modul tidak konsisten dengan proses bisnis yang sedang berjalan, baris program, variabel dan parameter yang terlibat dalam unit akan diperiksa dan dikoreksi satu per satu, dan kemudian dikompilasi ulang.

2.9.4 *Black box Testing*

Fokus pada persyaratan fungsional perangkat lunak. Oleh karena itu, pengujian kotak hitam memungkinkan pengembang perangkat lunak untuk membuat sekumpulan kondisi input untuk mengimplementasikan semua persyaratan fungsional program. Pengujian black box bukanlah alternatif pengujian white box, tetapi metode pelengkap untuk menemukan kesalahan selain metode white box.

2.10 *Open Source*

Menurut (kominfo, 2013) Open source adalah sistem pengembangan yang tidak dikoordinasikan oleh individu / instansi pusat, tetapi oleh peserta yang bekerja sama melalui distribusi dan penggunaan kode sumber secara gratis (biasanya

menggunakan alat komunikasi internet). Model pengembangan ini menggunakan model bazaar style, sehingga model open source ini memiliki keunikan tersendiri yaitu dorongan dari budaya donasi, artinya ketika komunitas menggunakan program open source dan mendapatkan keuntungan, mereka akan termotivasi untuk melamar. satu pertanyaan. Pengguna dapat memberikan kembali kepada orang banyak.

Model open source lahir karena memiliki kebebasan untuk bekerja, tanpa memikirkan intervensi dan menggunakan pengetahuan dan produk yang sesuai untuk mengekspresikan apa yang diinginkan. Ketika kebebasan diumumkan kepada publik, kebebasan menjadi pertimbangan utama. Komunitas lain memiliki kebebasan untuk belajar, mengutak-atik, memodifikasi, membenarkan bahkan menyalahkan, tetapi kebebasan ini juga hadir dengan tanggung jawab, bukan kebebasan tanpa tanggung jawab.

2.11 *Flowchart*

Saat mendesain diagram alur, sebenarnya tidak ada (beberapa) rumus atau standar absolut. Hal tersebut dapat didasarkan pada diagram alur (flow chart) yang merupakan gambaran hasil berpikir pada saat menganalisis suatu masalah di komputer. Karena setiap analisa akan menghasilkan hasil yang berbeda. Meski begitu, secara garis besar setiap desain diagram alir akan selalu terdiri dari tiga bagian yaitu input, proses dan output. Flowchart juga memiliki simbol-simbol yang memiliki arti tersendiri dalam proses pembangunan sistem. Sebagai analis sistem, diagram alur ini digunakan sebagai dasar untuk dapat mengirimkan program kepada programmer atau pengembang

BAB III

ANALISIS MASALAH DAN PERANCANGAN PROGRAM

3.1 Analisis Masalah pada Penelitian Sebelumnya

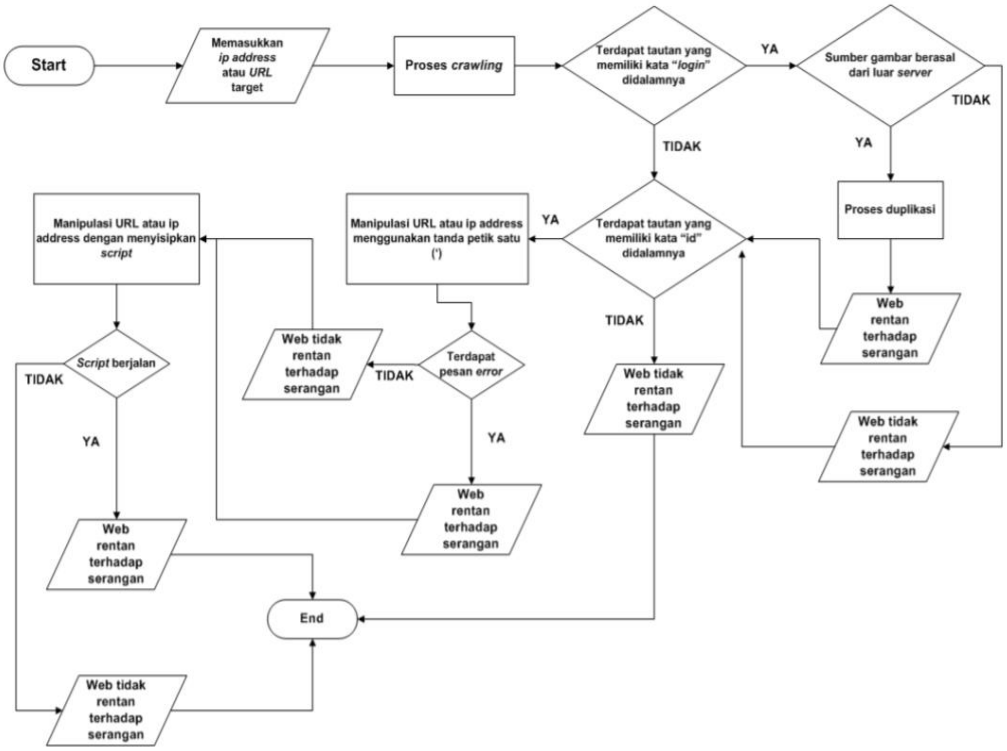
Pada penelitian sebelumnya yaitu (Yudha. F., Panji. A.M. 2018) melakukan Perancangan Aplikasi Pengujian Celah Keamanan Pada Aplikasi Berbasis *Web*, aplikasi yang dibangun untuk mendeteksi celah keamanan pada jenis serangan *SQL Injection, Cross-Site Script* dan *phising* tetapi masih terdapat kekurangan pada metode pendeteksian celah keamanan yaitu :

1. Metode *crawling* hanya bisa menelusuri pada halaman depan
2. Metode *SQL Injection* dan *Cross-Site Script* menguji pada website yang terdapat page "*id*"

Sehingga menjadi kurang efektif dalam pendeteksian kerentanan pada *website* yang akan diuji.

3.1.1 *Flowchart* pada Penelitian Sebelumnya

Pada gambar 3.1 adalah *flowchart* metode scan dari penelitian yang dilakukan oleh (yuhda. F., Panji. A.M. 2018)



Gambar 3.1 flowchart metode scan (Yudha. F., Panji. A.M. 2018)

3.2 Analisis Sistem yang akan dibangun

3.2.1 Deskripsi Umum Sistem yang Akan Dibangun

Aplikasi yang akan dibangun pada tugas akhir ini adalah Perancangan Aplikasi Deteksi Kerentanan *SQL Injection* dan *Cross-Site Script Website* Menggunakan Metode *Waterfall*

3.2.2 Analisis kebutuhan Aplikasi

1. Aplikasi yang dibangun bisa melakukan *crawling* terhadap semua konten yang ada pada *website* target.
2. Metode pendeteksian kerentanan *SQL Injection* dan *Cross-Site Script* dengan menginputkan pada *form* input *HTML*.

3.2.3 Analisis Kebutuhan Perangkat Lunak dan Perangkat keras

Dalam pembuatan aplikasi ini dibutuhkan beberapa perangakat lunak dan perangkat keras yang mendukung jalannya aplikasi ini antara lain :

A. Kebutuhan Perangkat Lunak (software)

Perangkat lunak yang digunakan untuk membangun aplikasi ini antara lain:

1. XAMPP yang di dalamnya terdapat *web server apache, php 5.3* dan MySQL sebagai *database,*
2. *Notepad++* sebagai text editor yang digunakan untuk pembuatan source code pada aplikasi yang dibuat;
3. *Google chrome* sebagai *web client* yang menampilakn user interface dari aplikasi yang dibuat;
4. Visual Paradigm sebagai tools untuk merancang pembuatan diagram dan grafik fector untuk mendeskripsikan suatu proses bisnis;
5. Operating System windows 10;

B. Kebutuhan Perangkat Keras (hardware)

Perangakat keras yang digunakan untuk membangun aplikasi ini adalah laptop atau desktop dengan spesifikasi minimum :

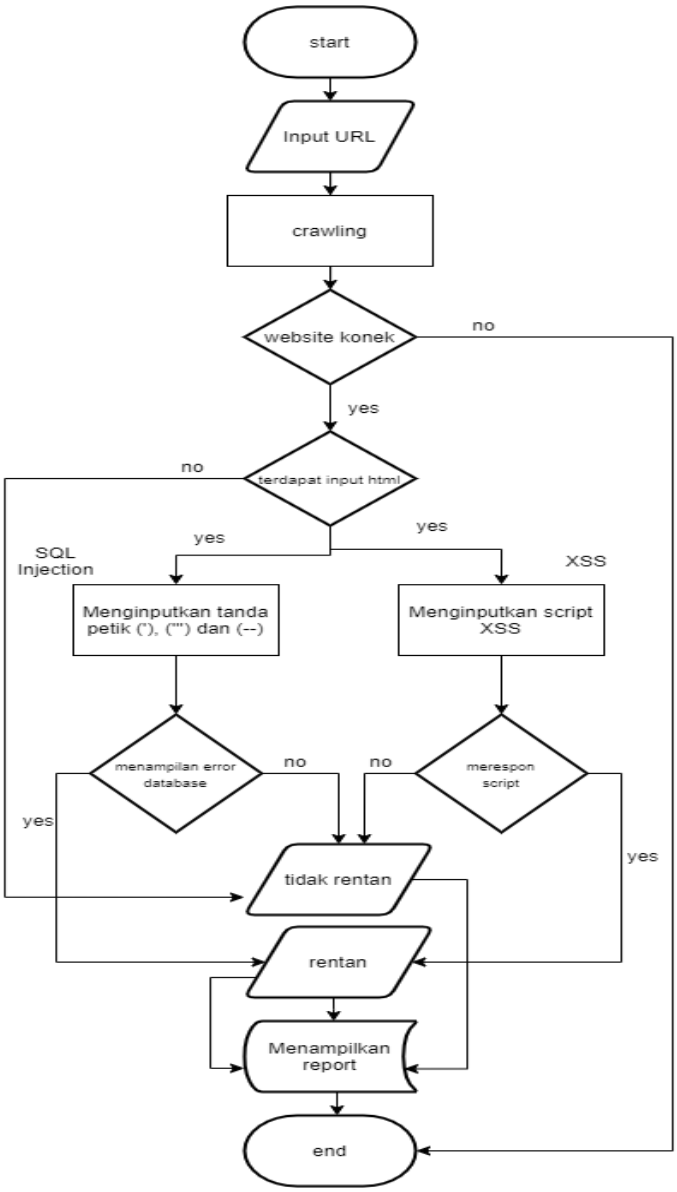
1. Processor : intel Pentium 4
2. Memory : 1 GB ram
3. Harddisk : 20 GB
4. Monitor : 10.0” inch
5. Hardware pendukung lainnnya: *Mouse, Keyboard. Lancard atau wifi*

3.3 Perancangan Aplikasi

Perancangan aplikasi dilakukan untuk menggambarkan bagaimana aplikasi dibentuk. Berupa perencanaan dan pembuatan sketsa program aplikasi.

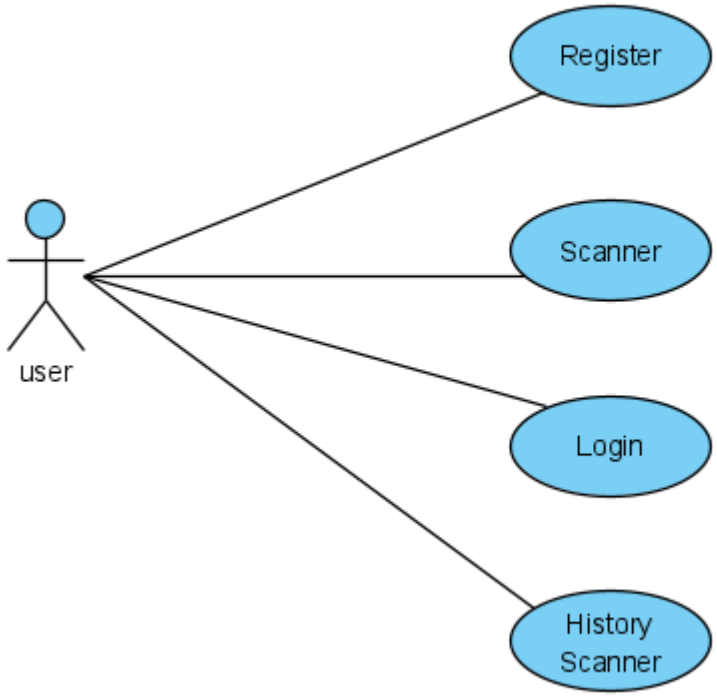
3.3.1 *Flowchart*

Pada gambar 3.2 adalah proses alur deteksi celah keamanan *website* , dimana dimulai dengan menginputkan alamat url target aplikasi akan melakukan *crawling* jika respon *website* tidak konek proses scanner akan berhenti, jika bukan maka selanjutnya aplikasi akan mencoba melakukan scan dengan metode yang ada yaitu *SQL Injection, Cross-Site Script*, ketika ada url yang terdeteksi kerentanan maka akan menampilan kerentanan tersebut.



Gambar 3.2 *Flowchart* alur deteksi keamanan

3.3.2 *Use Case Diagram*



Gambar 3.3 *Use Case Diagram*

3.3.2.1 *Use Case* register

Use case Register

Nama use case : Register

Actor : User

Tujuan : Melakukan pendaftaran user

Deskripsi : Aktor masuk ke halaman register dan memasukan Email dan password

Tabel 3.1 Use Case Register

Action	System user
Description	User memilih tombol register
Actor	user
Goal	Berhasil mendaftarkan user
Pre-Condition	Pilih tombol register dan isi form registrasi

Post-condition	User telah terdaftar

3.3.2.2 *Use Case* login

Use case login

Nama use case : login

Actor : User

Tujuan : Verivikasi Email dan password yang sudah terdaftar

Deskripsi : Aktor masuk kehalaman login dan memasukan Email dan password kemudian masuk ke menu utama

Tabel 3.2 Use Case login

Action	System user
Description	User memilih button login
Actor	user
Goal	Berhasil masuk ke halaman utama
Pre-Condition	Memasukan *username* dan *password*
Post-condition	Menampilkan Menu utama

3.3.2.3 *Use Case* Scanner

Use case scanner

Nama use case : Scanner

Actor : User

Tujuan : Melakukan pengujian alamat target

Deskripsi : Aktor masuk ke halaman scanner dan menginputkan url yang akan di scan

Tabel 3.3 *Use Case Scanner*

Action	System user
Description	User memilih menu scanner
Actor	user
Goal	Bisa melakukan scan url website
Pre-Condition	Menginputkan url alamat target

Post-condition	Report hasil scanner

3.3.2.4 *Use Case History*

Use case History

Nama use case : History

Actor : User

Tujuan : Mengecek hasil scanner

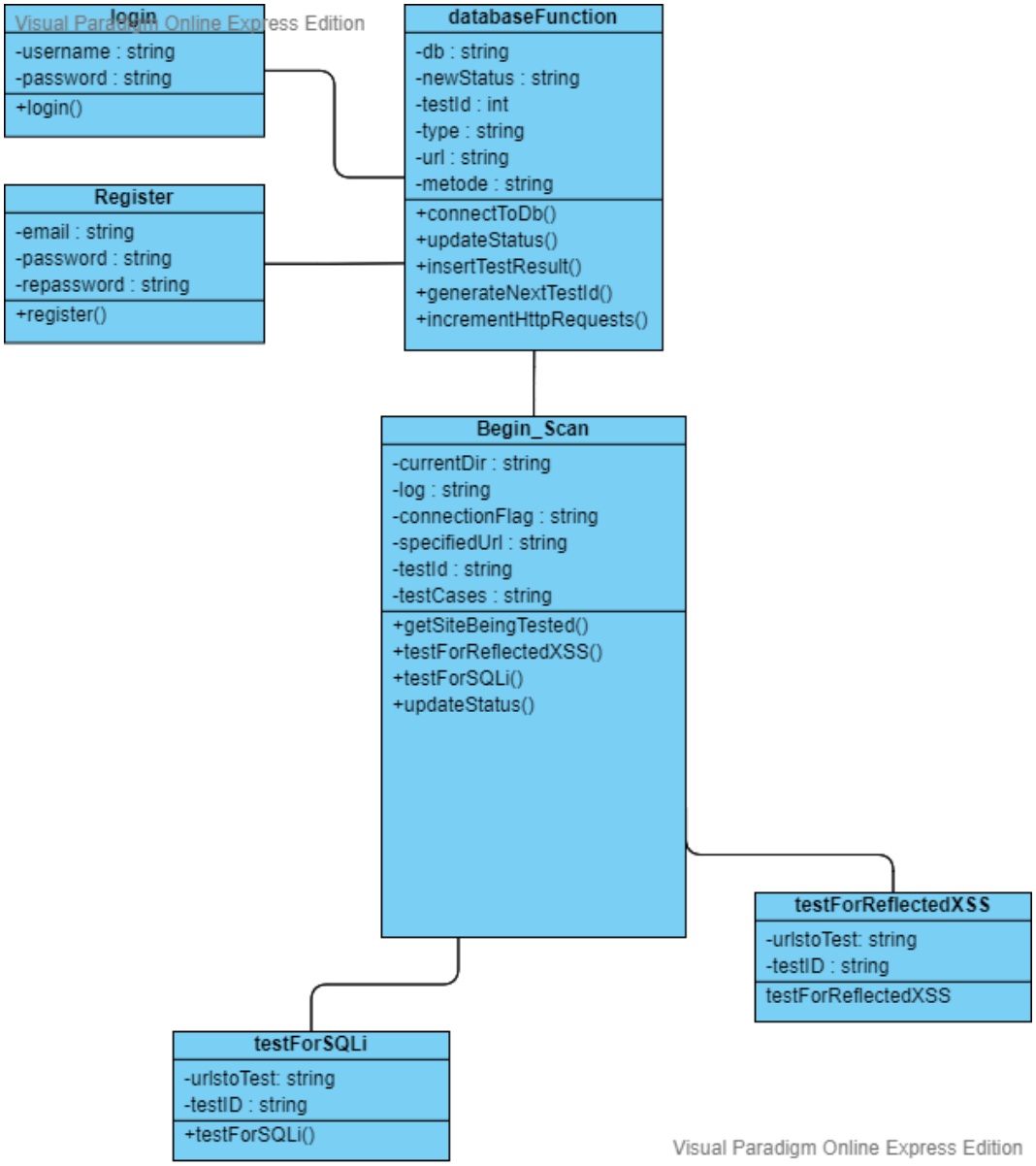
Deskripsi : Aktor masuk kehalaman history dan menampilkan report scanner

Tabel 3.4 *Use Case History Scanner*

Action	System user
Description	User memilih menu hasil scan
Actor	user
Goal	Mendapatkan informasi berupa report
Pre-Condition	Hasil scan web sudah tersedia
Post-condition	Berhasil mengambil data report hasil scan

3.3.3 Class Diagram

class diagram pada perancangan aplikasi yang akan dibanungun pada gambar 3.4 menampilkan hubungan antara kelas-kelas yang ada



Gambar 3.4 *Class Diagram*

3.3.4 *Activity Diagram*

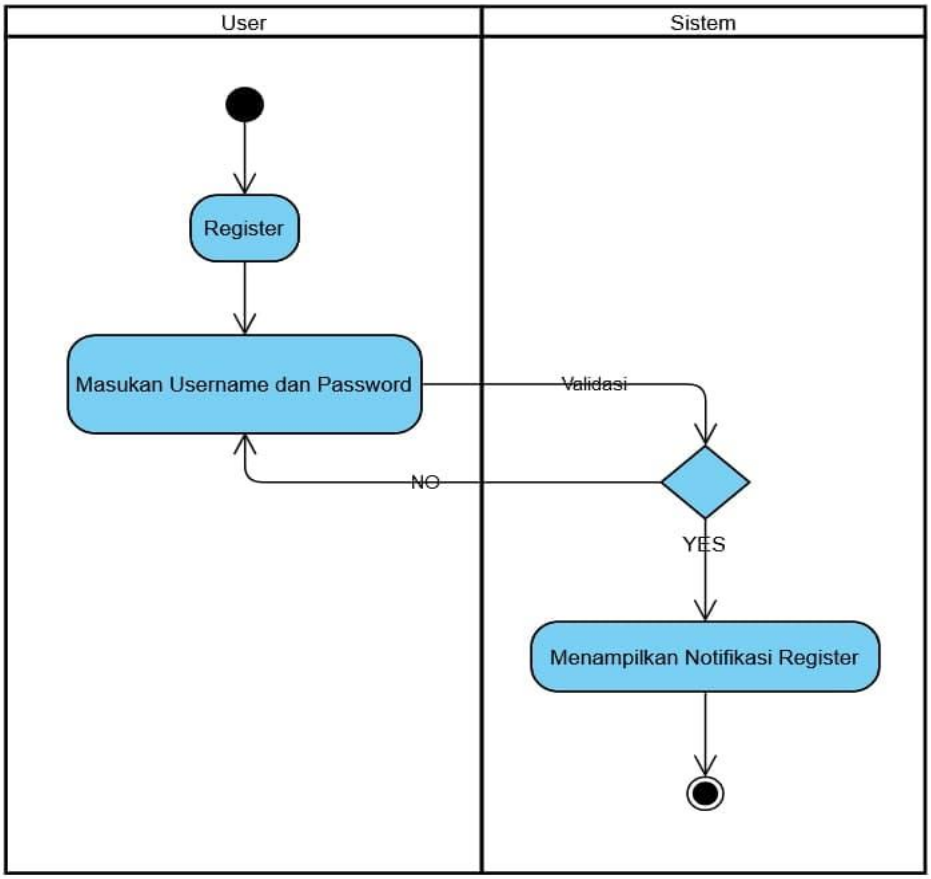
3.3.4.1 *Activity Diagram register*

Activity diagram register

Activity diagram : Register

Actor : User

Deskripsi : Aktor Memilih menu register lalu Memasukan username (email) dan password , confirm password dan email dan sistem akan memvalidasi apabila sudah selesai maka akan menampilkan pesan berhasil register



Gambar 3.5 *Activity Diagram register*

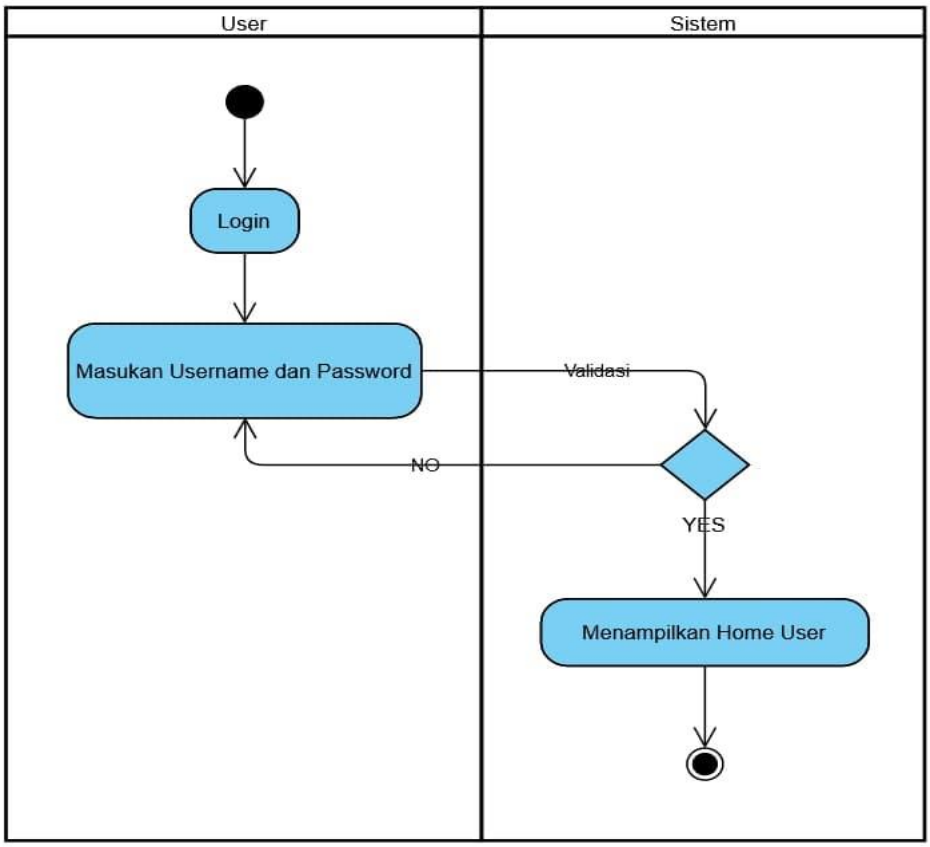
3.3.4.2 *Activity Diagram Login*

Activity diagram login

Activity diagram : login

Actor : User

Deskripsi : Aktor Memilih menu login lalu Memasukan username (email) dan password setelah itu di oleh sistem memvalidasi Username dan password, jika benar akan masuk ke menu utama, dan jika tidak benar menampilkan pesan kesalahan



Gambar 3.6 *Activity Diagram login*

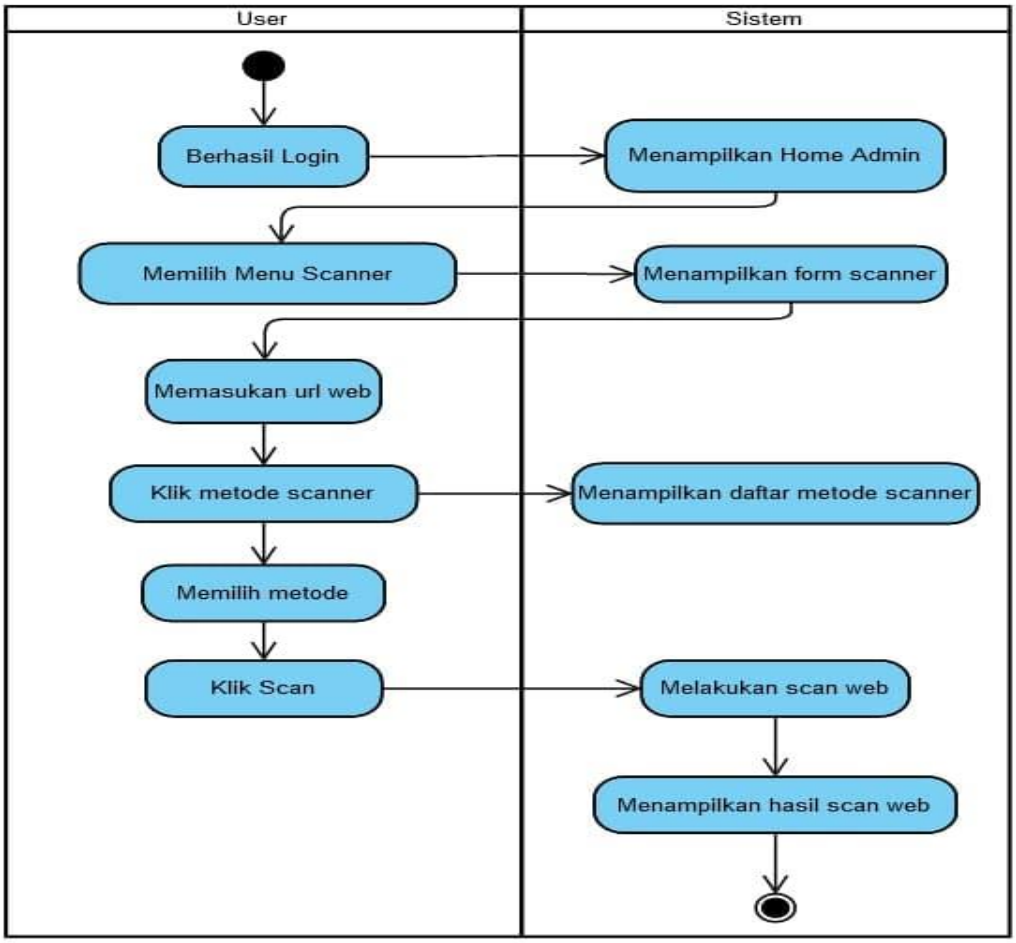
3.3.4.3 *Activity Diagram Scanner*

Activity diagram scanner

Activity diagram : scanner

Actor : User

Deskripsi : Aktor Memilih menu scanner lalu Aplikasi akan menampilkan form scanner dan user tinggal memasukan url alamat target dengan memilih metode scanner dan klik tombol start untuk memulai scanner nya, scanner sedang diproses dan ketika ditemukan kerentanan akan ditampilkan hasinya.



Gambar 3.7 *Activity Diagram scanner*

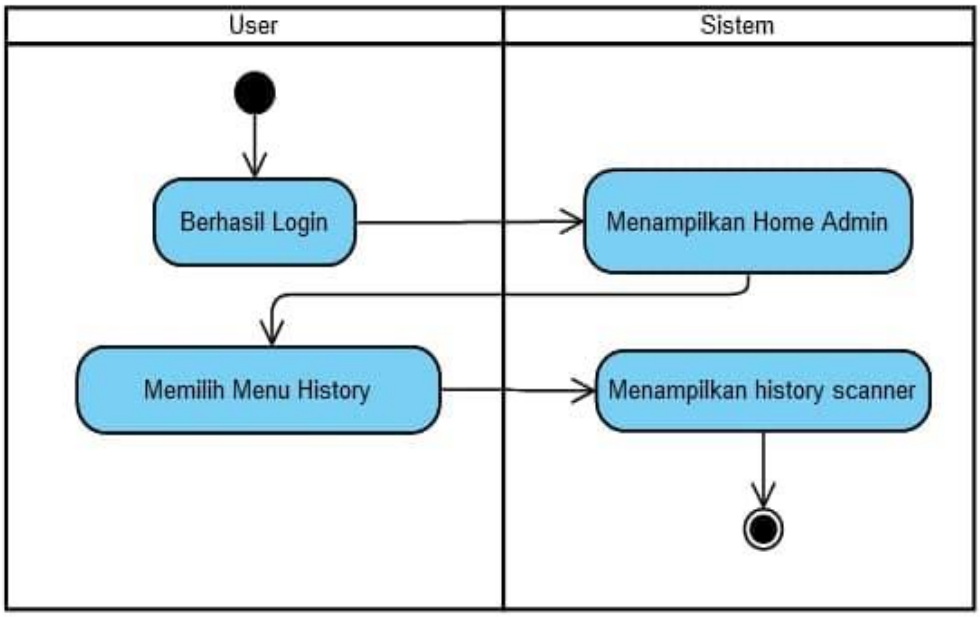
3.3.4.4 *Activity Diagram History*

Activity diagram history

Activity diagram : History

Actor : User

Deskripsi : Aktor login terlebih dahulu lalu sistem akan menampilkan menu home dan user melilih menu history dan sistem akan menampilkan history scanner.



Gambar 3.8 *Activity Diagram History*

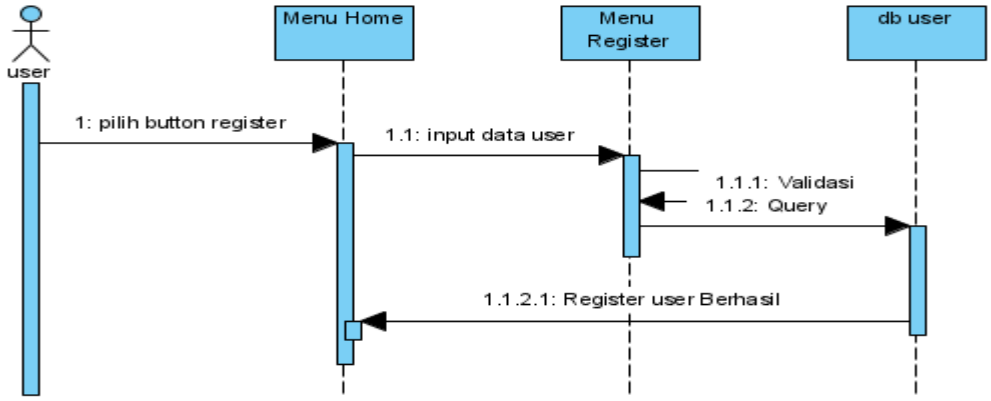
3.3.5 Sequence Diagram

Menggambarkan perilaku objek dalam use case dengan menjelaskan kehidupan objek dan pesan yang dikirim dan diterima di antara objek

3.3.5.1 *Sequence Diagram Register*

Sequence Diagram Register pada gambar 3.9

1. user diharuskan untuk melakukan register terlebih dahulu pada menu home lalu pilih login / register
2. sistem akan memvalidasi email dan password yang sudah dimasukan
3. jika sudah benar maka user akan di simpan pada database
4. Sistem akan menampilkan pesan registrasi telah berhasil

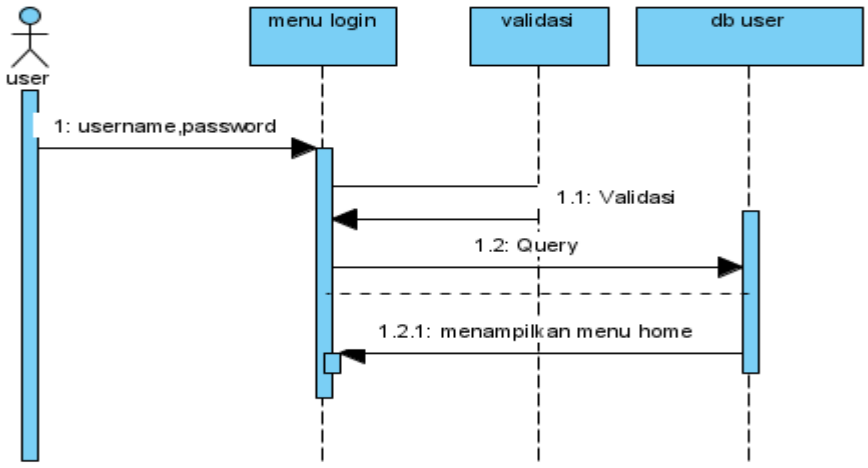


Gambar 3.9 *Sequence Diagram Register*

3.3.5.2 *Sequence Diagram Login*

Sequence Diagram login pada gambar 3.10

1. user diharuskan untuk memaskukan Email dan password kedalam menu login
2. sistem akan memvalidasi email dan password yang sudah dimasukan dengan yang ada di databse
3. jika sudah benar maka user akan di alihkan ke menu home

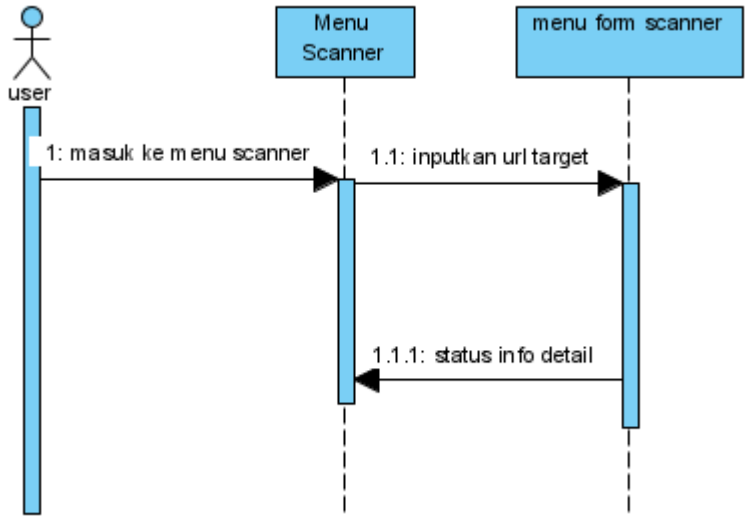


Gambar 3.10 *Sequence Diagram Login*

3.3.5.3 *Sequence Diagram Scanner*

Sequence Diagram Scanner pada gambar 3.11

1. user memilih *menu scanner*
2. sistem akan akan menampilkan *form scanner*
3. user menginputkan url target pada *form* yang sudah disediakan
4. sistem akan menampilkan info detail tentang proses dan hasil *scanner*

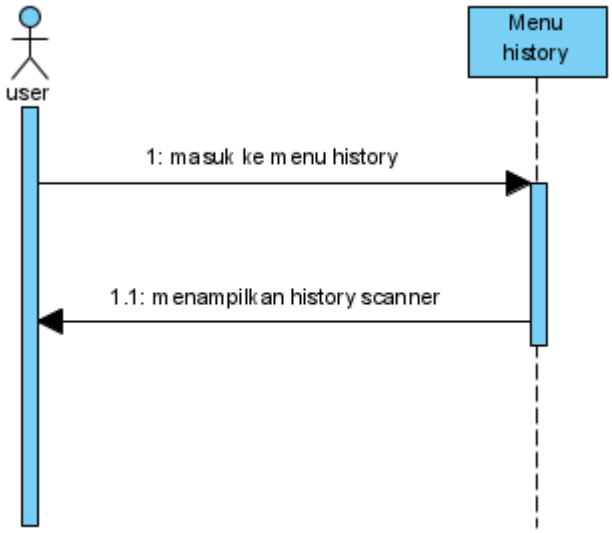


Gambar 3.11 Diagram sequence Scanner

3.3.5.4 *Sequence Diagram History*

Sequence Diagram history pada gambar 3.11

1. user memilih menu history
2. sistem akan akan menampilkan menu history
3. user bisa melihat seluruh hasil scanner



Gambar 3.12 *Sequence Diagram History Scanner*

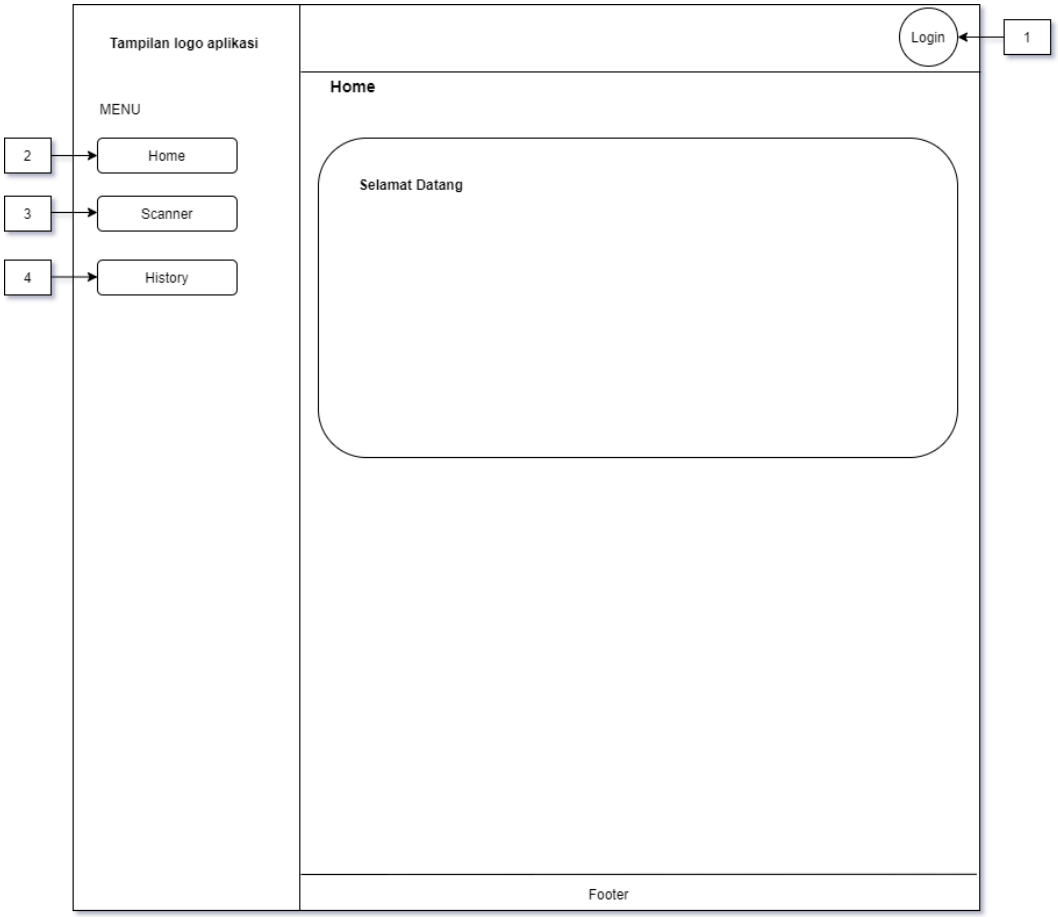
3.4 Perancangan Antarmuka (*Interface*)

Interface merupakan salah satu media yang digunakan untuk komunikasi antara manusia dengan komputer, sehingga tujuan dari perancangan aplikasi ini adalah untuk memudahkan pengguna aplikasi ini dalam menggunakan atau mengoperasikannya.

Beberapa tampilan *interface* pada beberapa menu dalam aplikasi apotek ini memiliki kesamaan format tampilan yang hampir mirip. Namun yang membedakannnya adalah hanya beberapa item tertentu.

Berikut contoh beberapa *interface* pada aplikasi yang dirancang :

3.4.1 Interface Menu Home

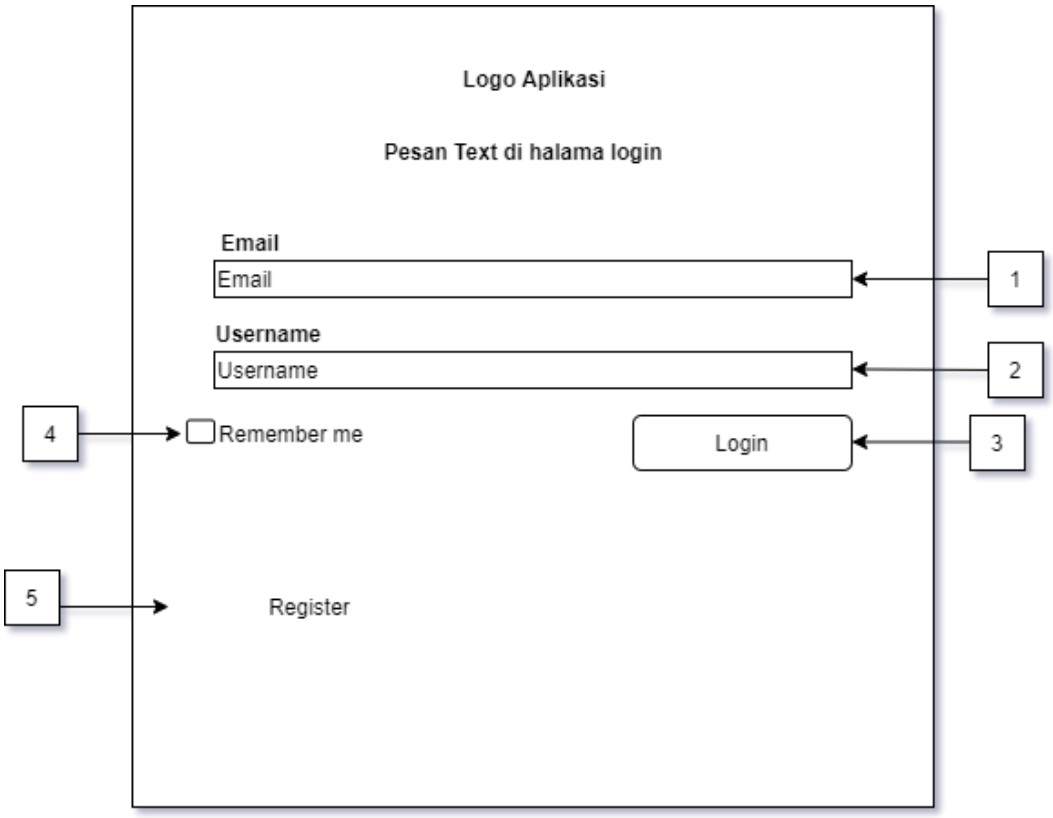


Gambar 3.13 *Interface Menu Home*

Keterangan Pada gambar 3.13 adalah :

1. Tombol untuk login
2. Menu home
3. Menu untuk scanner target
4. Menu untuk melihat history scanner

3.4.2 Interface Form Login

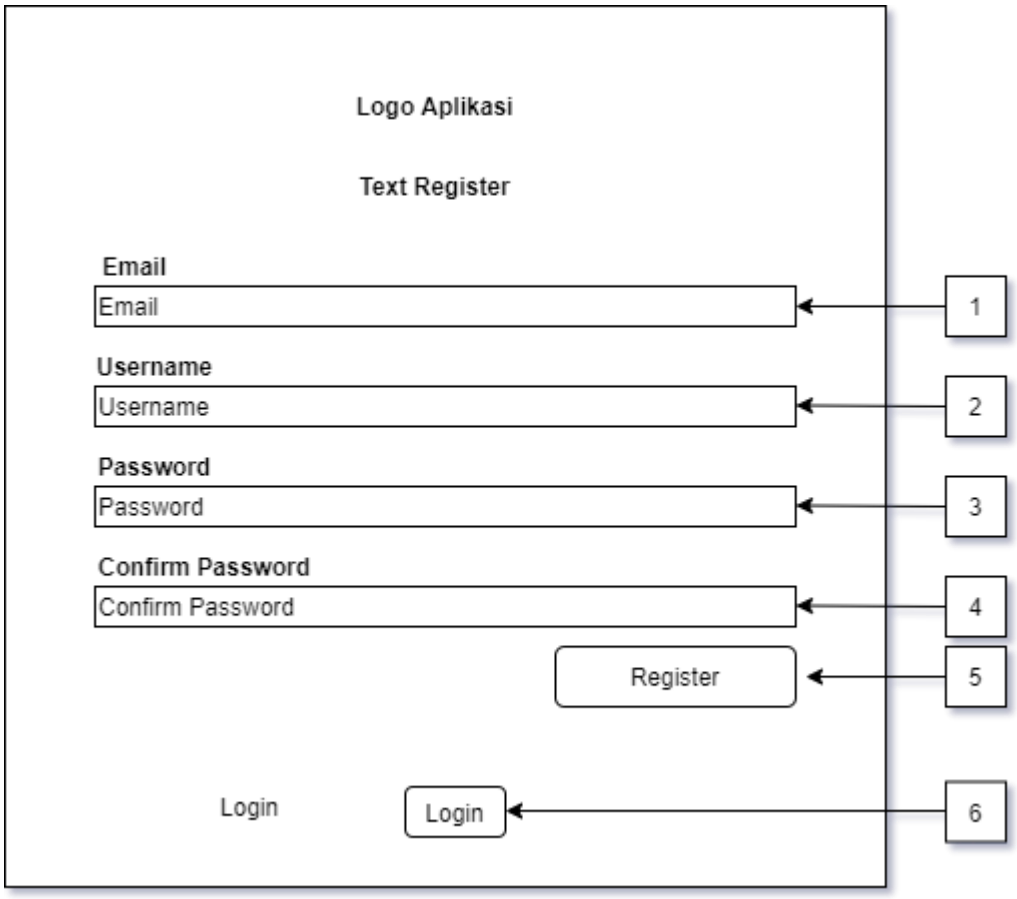


Gambar 3.14 *Interface Form Login*

Keterangan Pada gambar 3.14 sebagai berikut :

1. Untuk menginputkan Email
2. Untuk menginputkan username
3. Tombol untuk melanjutkan Login
4. Centang untuk remember Email dan Password sehingga otomatis tersimpan
5. Menu untuk mengarahkan ke halaman register

3.4.3 Interface Form Register

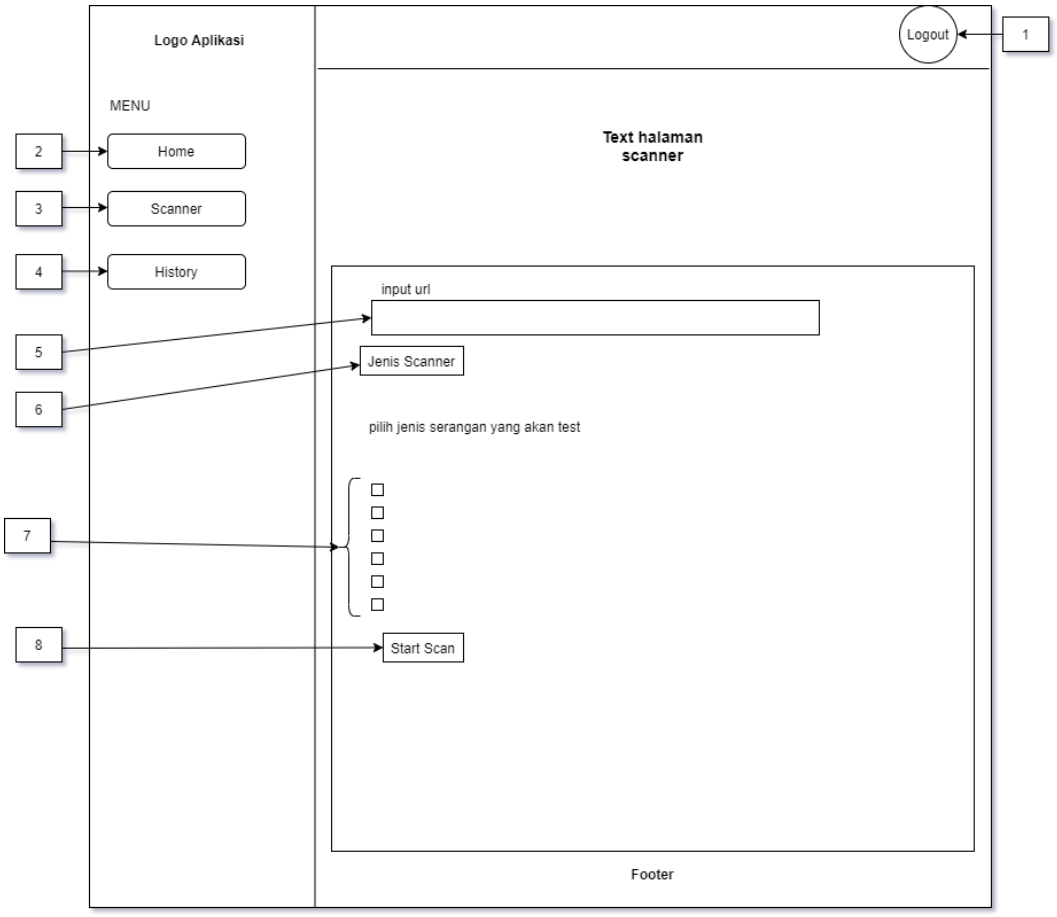


Gambar 3.15 *Interface Form Register*

Keterangan Pada gambar 3.15 sebagai berikut :

1. Untuk menginputkan Email
2. Untuk menginputkan username
3. Untuk menginputkan password
4. Untuk mengulang konfirmasi password
5. Tombol untuk meneruskan register
6. Tombol menu login jika sudah terdaftar

3.4.4 Interface Form Scanner

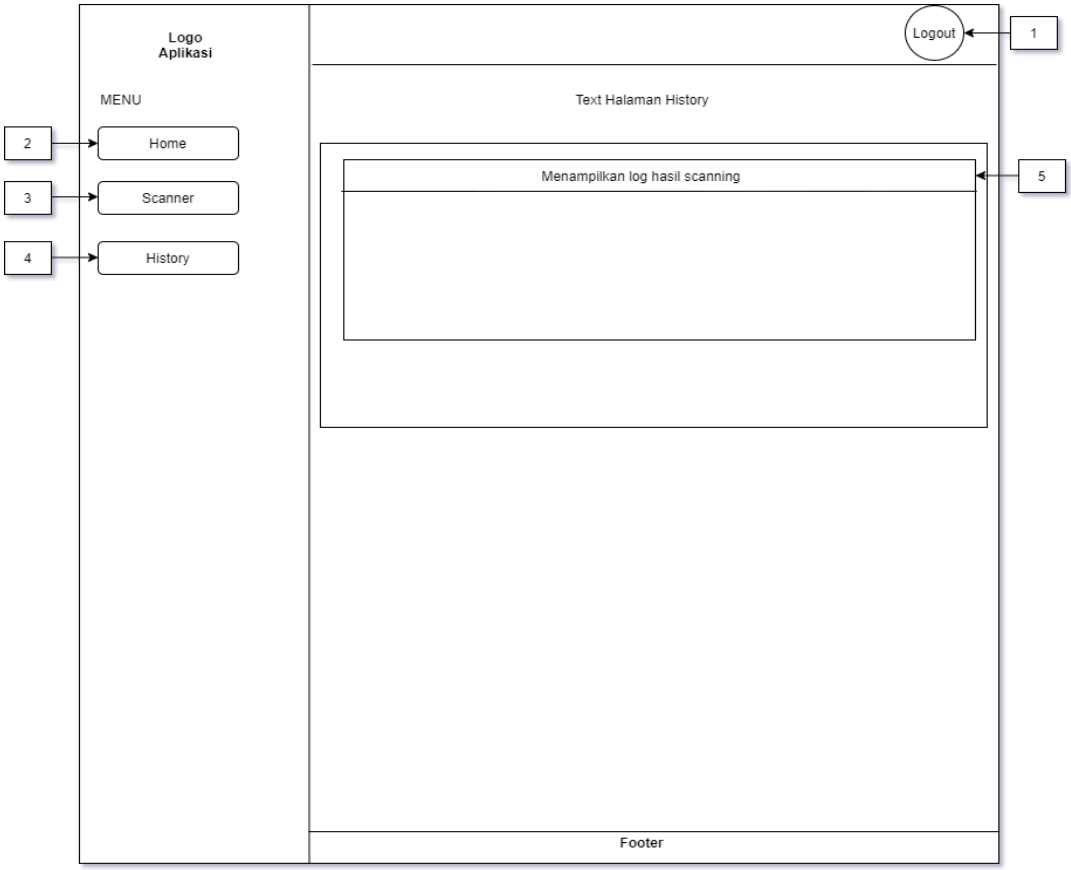


Gambar 3.16 Interface *Form Scanner*

Keterangan Pada gambar 3.16 sebagai berikut :

1. Tombol untuk logout
2. Tombol menu home
3. Tombol menu scanner
4. Tombol menu history
5. Form Scanner untuk menginputkan alamat url untuk dilakukan pengujian
6. Tombol untuk metode serangan
7. Menu pilihan metode serangan
8. Tombol untuk memulai proses pengujian

3.4.5 Interface Menu History



Gambar 3.17 *Interface Menu History*

Keterangan Pada gambar 3.17 sebagai berikut :

1. Tombol untuk logout
2. Tombol menu home
3. Tombol menu scanner
4. Tombol menu history
5. Halaman History scanner

BAB IV

IMPLEMENTASI DAN UJI COBA

4.1 Implementasi Aplikasi

Tahap implementasi merupakan tahap penerapan dari hasil analisis dan perancangan sistem sebelumnya.

4.1.1 Implementasi Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan dalam implementasi aplikasi ini adalah sebagai berikut:

1. XAMPP di dalamnya sudah ada *web server* apache, php 5.3 dan MySQL;
2. Notepad ++ sebagai text editor yang digunakan untuk pembuatan source code pada aplikasi yang dibuat;
3. Google chrome sebagai web client yang menampilakn user interface;
4. Visual Paradigm sebagai tools online untuk merancang pembuatan diagram.
5. Operating System windows 10;

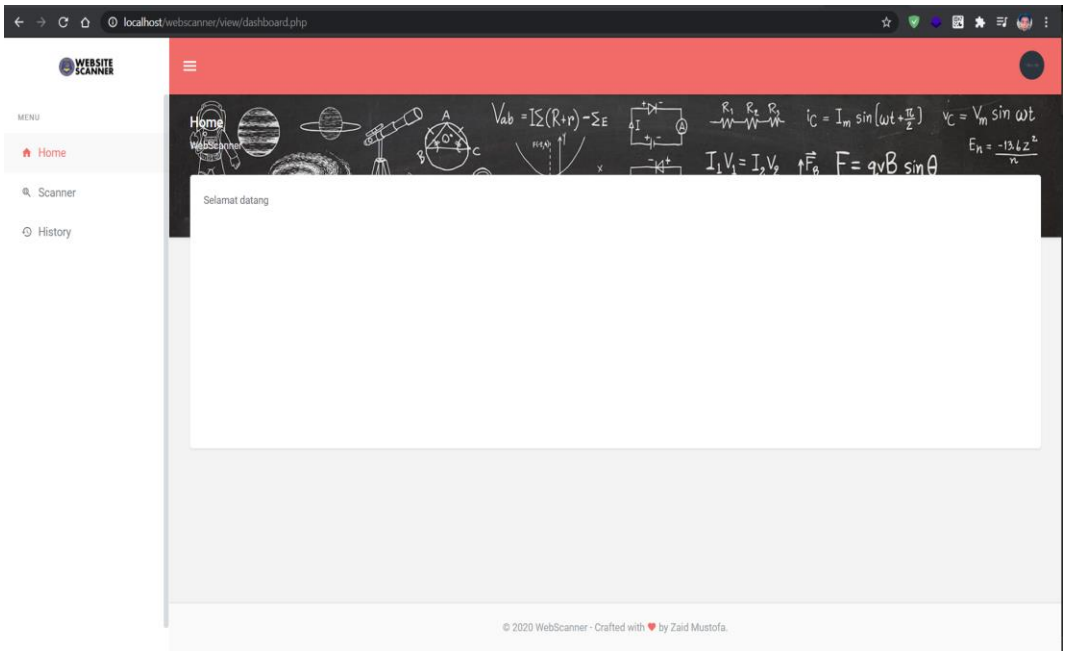
4.1.2 Implementasi Perangkat Keras

Implementasi perangkat keras merupakan salah satu syarat untuk implementasi perangkat lunak dan akan dilakukan pada tahap selanjutnya. Spesifikasi perangkat keras PC (komputer pribadi) yang digunakan adalah sebagai berikut :

1. Processor : intel core i5-8265U CPU @ 1.60Ghz (8 CPus), ~1.8Ghz
2. Memory : 20GB ram
3. Harddisk : ssd nvme 500GB
4. Monitor : 14.0” inch LED FHD 1080
5. Hardware pendukung lainnnya: *Mouse, Keyboard. Lancard, wifi*

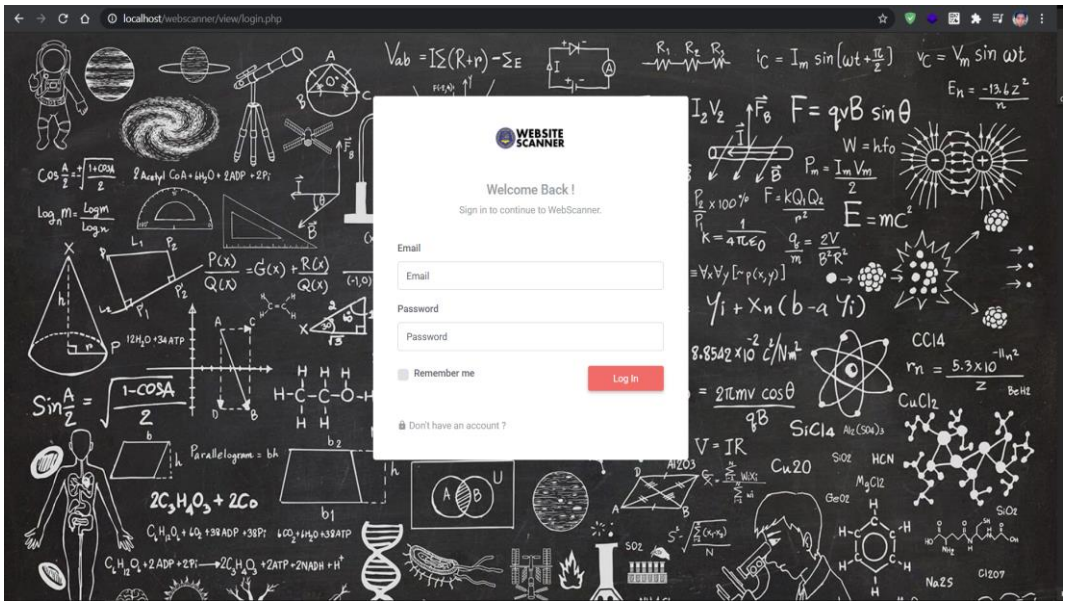
4.1.3 Implementasi Antar Muka (*Interface)*

4.1.3.1 *Menu Home*



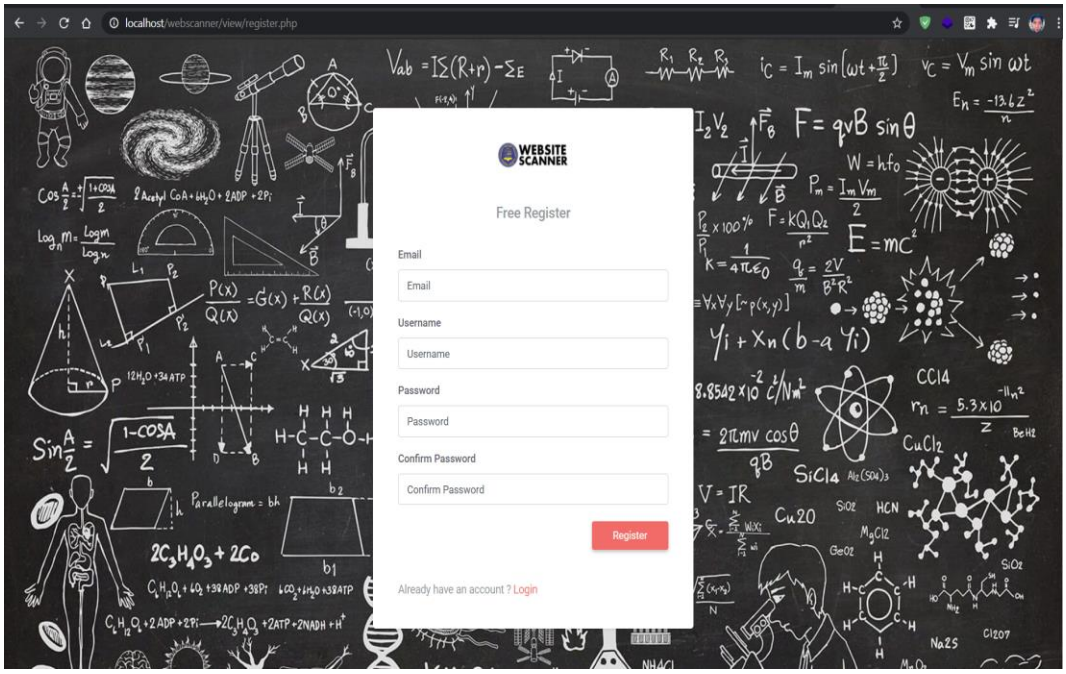
Gambar 4.1 *menu home*

4.1.3.2 *Menu Form login*



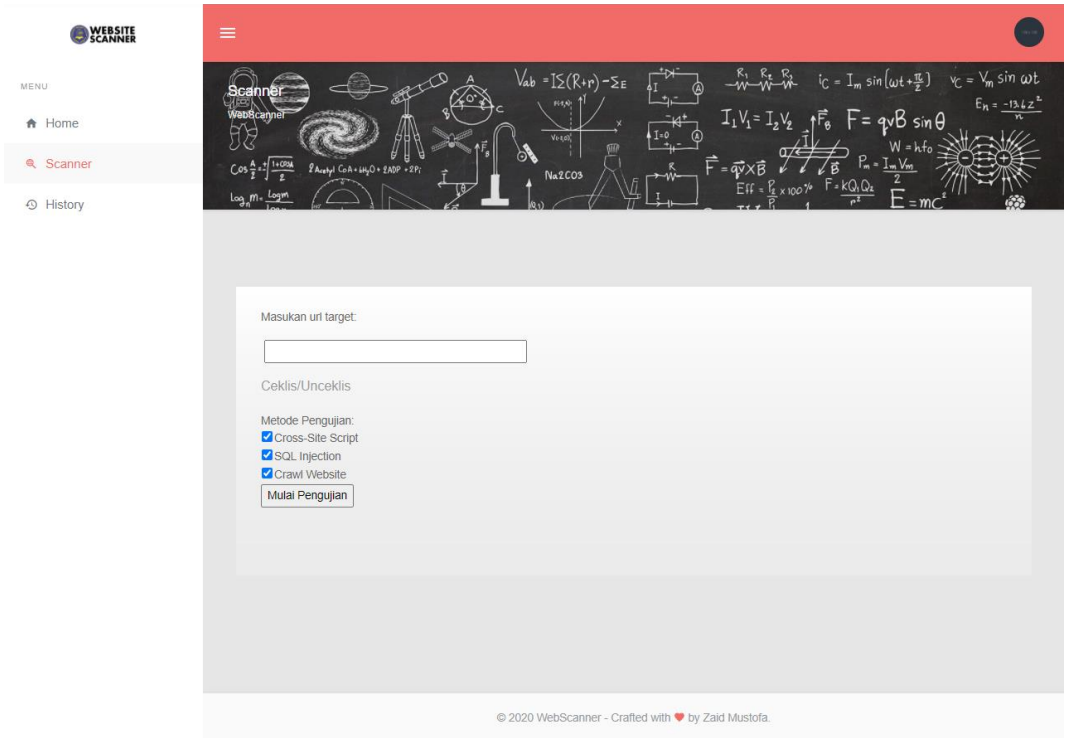
Gambar 4.2 *Menu Form login*

4.1.3.3 *Menu Form Register*



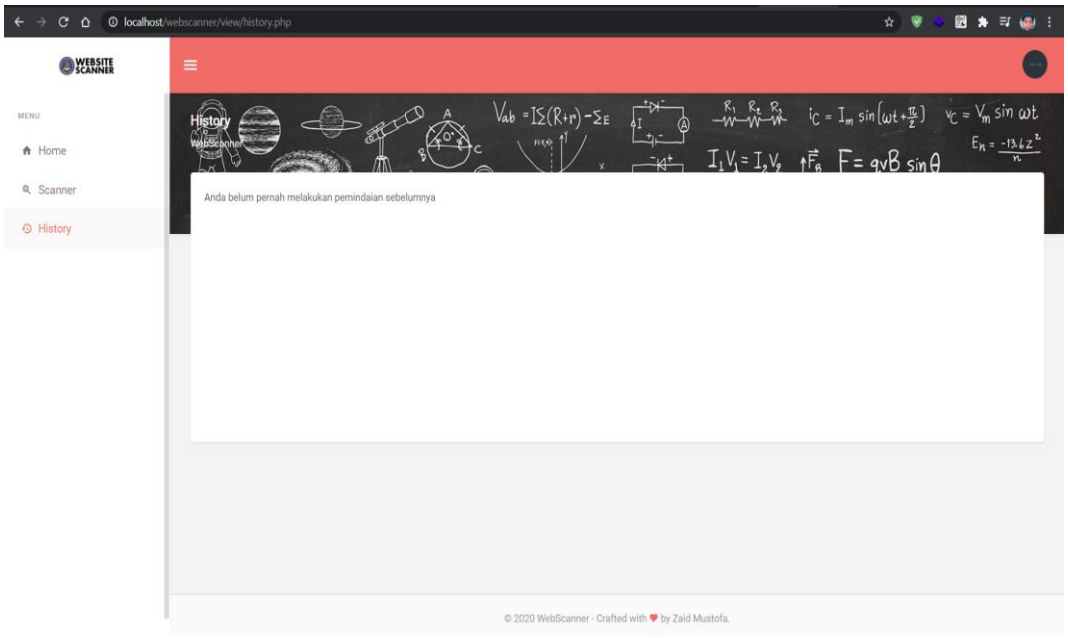
Gambar 4.3 *Menu Form Register*

4.1.3.4 *Menu Form scanner*



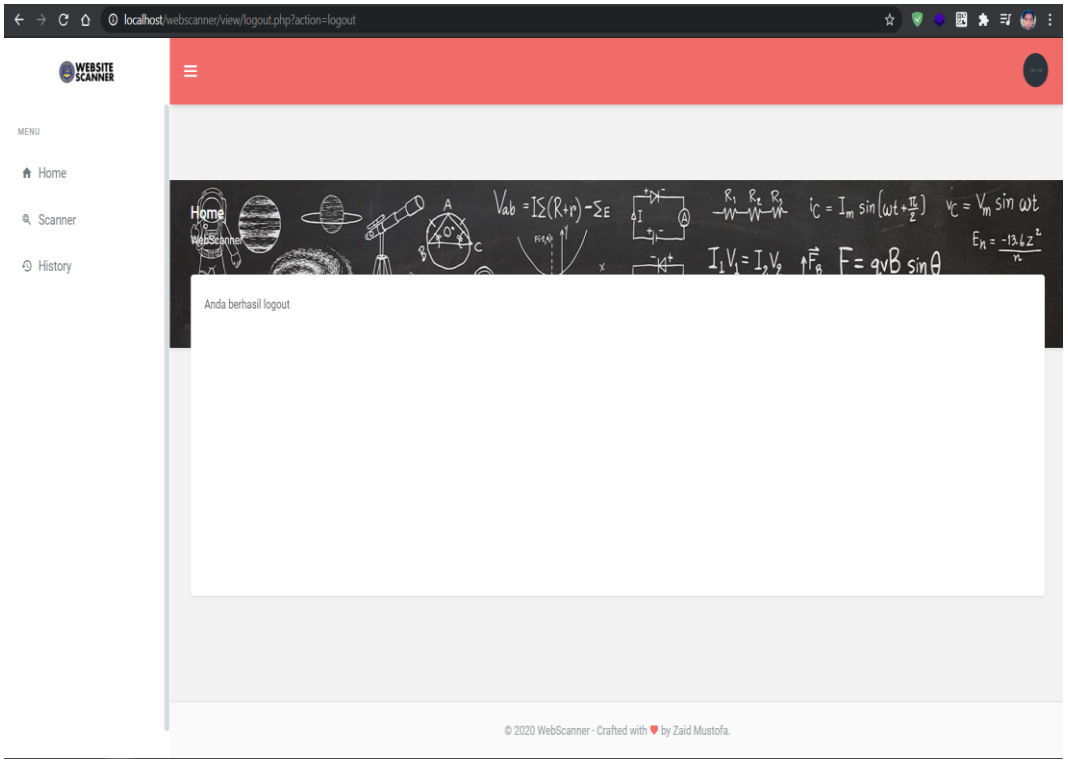
Gambar 4.4 *Menu Form scanner*

4.1.3.5 *Menu History*



Gambar 4.5 *Menu History*

4.1.3.6 *Menu* setelah *Logout*



Gambar 4.6 *Menu* setelah *Logout*

4.2 Pengujian Aplikasi

4.2.1 Pengujian *Black Box*

Pengujian menggunakan metode *Black Box* pada antarmuka dan proses scan nya. Hasil pengujian yang dilakukan harus menunjukkan bahwa sistem telah beroperasi sesuai dengan tujuan penelitian dan beroperasi sesuai dengan yang diharapkan.

Black Box testing adalah adalah metode pengujian perangkat lunak yang digunakan untuk menguji fungsi dan struktur internal suatu aplikasi. Kasus uji dibuat berdasarkan spesifikasi dan persyaratan (yaitu, apa yang harus dilakukan aplikasi). Metode pengujian dapat diterapkan ke semua tingkatan pengujian perangkat lunak: unit, integrasi, fungsi, sistem, dan penerimaan. Ini biasanya mencakup sebagian besar (jika tidak semua) pengujian tingkat yang lebih tinggi, tetapi juga dapat mendominasi pengujian unit

Uji coba *Black Box* mencoba menemukan beberapa jenis kesalahan, termasuk :

1. Kesalahan fungsi *register*
2. Kesalahan fungsi *login*
3. Kesalahan fungsi *scanner*
4. Kesalahan log hasil scan

Berikut adalah pengujian yang dilakukan pada sistem dengan menggunakan metode blackbox *testing.*

Tabel 4.1 Hasil Pengujian *Blackbox menu Register*

Nama	Aktifitas	Input Data	Hasil perencanaan	Hasil Implementasi	Keterangan
Register	register	- Email : zaid@gmail.com - Username : zaid - Password : zaid123 - Confirm password : zaid123	- Menampilkan pesan Selamat! Anda berhasil mendaftar, Silahkan login untuk menggunakan aplikasi - User ter register di database	- Menampilkan pesan Selamat! Anda berhasil mendaftar, Silahkan login untuk menggunakan aplikasi - User ter register di database	Diterima
	Register jika ada username yang sama	- Email : zaid@gmail.com - Username : zaid - Password : zaid123 - Confirm password : zaid123	- Menampilkan pesan punten, username ini sudah terdaftar, silahkan menggunakan yang lain	- Menampilkan pesan punten, username ini sudah terdaftar, silahkan menggunakan yang lain	Diterima
	Register jika ada Email yang sama	- Email : zaid@gmail.com - Username : zaid - Password : zaid123 - Confirm password : zaid123	- Menampilkan pesan punten, Email ini sudah terdaftar, silahkan menggunakan yang lain	- Menampilkan pesan punten, Email ini sudah terdaftar, silahkan menggunakan yang lain	Diterima

Tabel 4.2 Hasil Pengujian *Blackbox Menu Login*

Nama	Aktifitas	Input Data	Hasil perencanaan	Hasil Implementasi	Keterangan
Login	Login user yang terdaftar	- Email : zaid@gmail.com - Password : zaid123	User diarahkan ke menu dashbaord	User diarahkan ke menu dashbaord	Diterima
	Username / Email salah	- Email : zaid@gmail.com - Password : zaid	Menampilkan pesan kesalahan Email atau Password. Silahkan periksa kembali	Menampilkan pesan kesalahan Email atau Password. Silahkan periksa kembali	Diterima
	Username / Email kosong	- Email : kosong - Password : kosong	Menampilkan pesan kesalahan Email atau Password. Silahkan periksa kembali	Menampilkan pesan kesalahan Email atau Password. Silahkan periksa kembali	Diterima
	Mencoba login ketika user sebelumnya sudah login	- Email : zaid@gmail.com - Password : zaid123	user dialihkan ke menu home	User tidak dialihkan ke menu home	Ditolak

Tabel 4.3 Hasil Pengujian Metode *Blackbox Menu Scanner*

Nama	Aktifitas	Input Data	Hasil perencaan	Hasil Implementasi	Keterangan
scanner	input alamat website target	ketikan url website target	Aplikasi akan otomatis mengecek kerentanan website target, Ketika ada kerentanan akan ditampilkan	Aplikasi mengecek kerentanan website target, Ketika ada kerentanan akan ditampilkan	Diterima
	input alamat website target	url target kosong	Menampilkan pesan error : Tidak ada URL yang di masukkan	Menampilkan pesan error : Tidak ada URL yang di masukkan	Diterima
	Akses menu scanner tanpa login		Menampilkan pesan : Sekarang anda belum login. Mohon untuk login terlebih dahulu	Menampilkan pesan : Sekarang anda belum login. Mohon untuk login terlebih dahulu	Diterima

Tabel 4.4 Hasil Pengujian *Blackbox Menu History*

Nama	Aktifitas	Input Data	Hasil perencaan	Hasil Implementasi	Keterangan
History	Akses menu History		Menampilkan log scanner	Menampilkan log scanner	Diterima
	Download / View Report		Menampilkan hasil report scanner pada target sebelumnya	Menampilkan hasil report scanner pada target sebelumnya	Diterima
	Akses menu History tanpa login		Menampilkan pesan : Sekarang anda belum login. Mohon untuk login terlebih dahulu	Menampilkan pesan : Sekarang anda belum login. Mohon untuk login terlebih dahulu	Diterima

4.2.2 Pengujian fungsi scanner

Tujuan dari tahap ini adalah untuk memastikan apakah hasil keluaran dari aplikasi sudah memenuhi ekspektasi atau belum. Sasaran pengujian berisi alamat *website* yang telah ditentukan sebelumnya.

Alamat target *website* dapat dilihat pada tabel 4.5

Tabel 4.5 Alamat Target

No	Alamat url	Keterangan web
1	http://testphp.vulnweb.com/	Web vuln acunetix
2	http://10.251.251.151	see lab
3	http://10.251.251.80	laak feb
4	http://10.251.251.194	Library
5	http://10.252.252.139	Openlib
6	http://10.251.251.72/	Smart parking
7	http://10.251.251.178	Aset
8	https://10.252.252.19/	Sister
9	https://118.xx.xx.118	web short

4.2.3 Target yang akan diuji

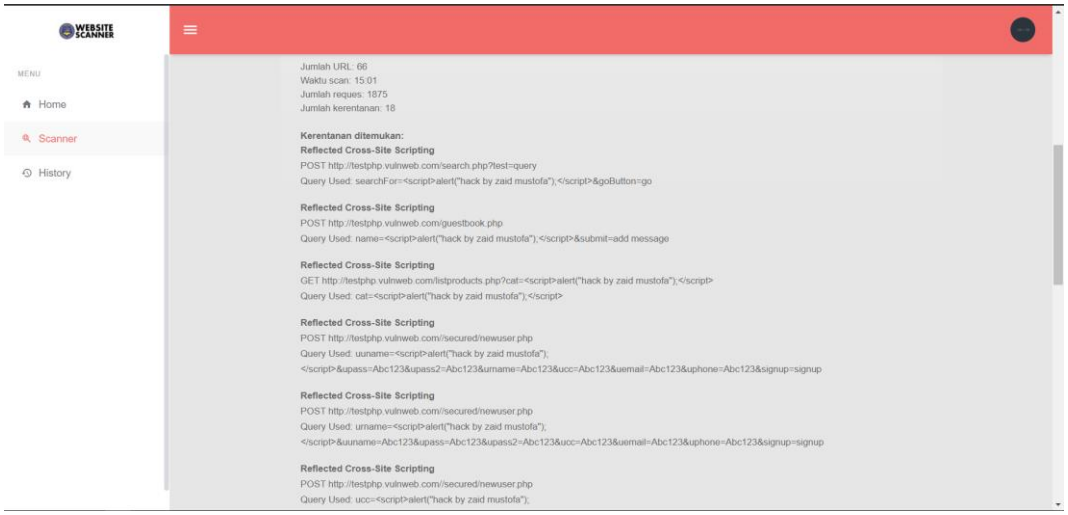
Pada tahap ini, proses pengujian pada fungsi *scanner*, dengan memilih menu *scanner* dan memasukan alamat url target. Proses pengujian terdapat beberapa metode yaitu *SQL Injection, Cross-Site Script :*

1. http://testphp.vulnweb.com

Website ini didesain untuk rentan terhadap semua jenis serangan, oleh karena itu jika aplikasi yang dirancang tidak bisa mendeteksi celah keamanan maka bisa dikatakan aplikasi tidak berfungsi sebagai mana mestinya.

untuk hasil pengujian nya bisa di lihat pada gambar 4.7 berikut ini :

Pada gambar 4.7 menampilkan proses scan pada alamat http://testphp.vulnweb.com

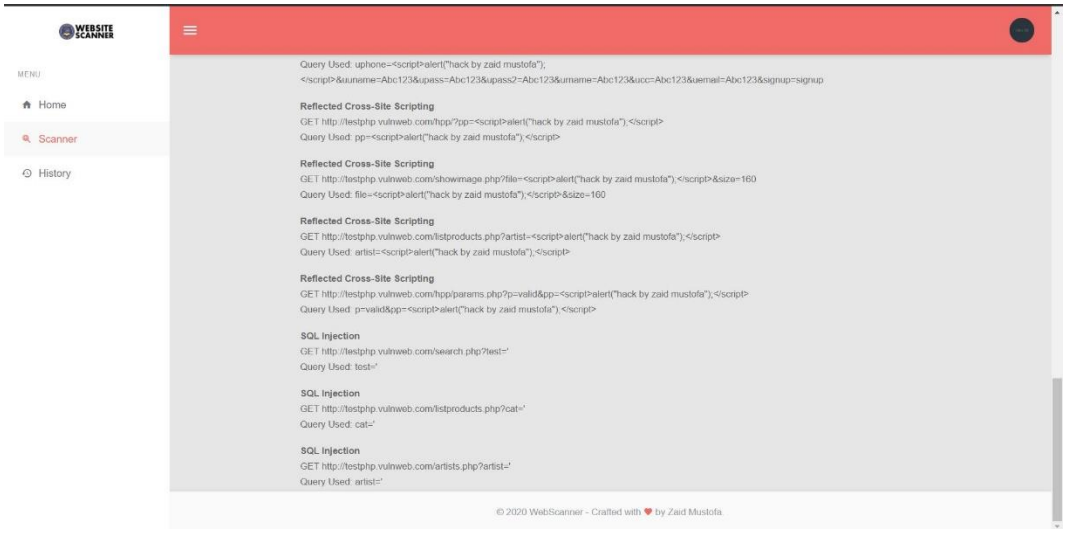


Gambar 4.7 Hasil Proses *scanner website* http://testphp.vulnweb.com

Untuk report scan Alamat http://testphp.vulnweb.com bisa dilihat pada gambar 4.8, 4.9:

Pada gambar 4.8 menampilkan bahwa *website* http://testphp.vulnweb.com memiliki kerentanan terhadap *SQL Injection,* yaitu diakhiran url tersebut dicoba di inputkan tanpa kutip satu

Pada gambar 4.9 menampilkan bahwa *website* http://testphp.vulnweb.com memiliki kerentanan terhadap *Cross-Site Script*, yaitu dengan menginputkan “<script>alert(“hack by zaid mustofa”);</script>” pada kolom *search.php, guestbook.php, listproducts.php,*



Gambar 4.8 Report: Serangan *SQL Injection* pada website testphp.vulnweb.com

Terletak pada:

URL: http://testphp.vulnweb.com/search.php?test=query
Method: POST
Query Used: searchFor=<script>alert("hack by zaid mustofa");</script>&goButton=go

URL: http://testphp.vulnweb.com/guestbook.php
Method: POST
Query Used: name=<script>alert("hack by zaid mustofa");</script>&submit=add message

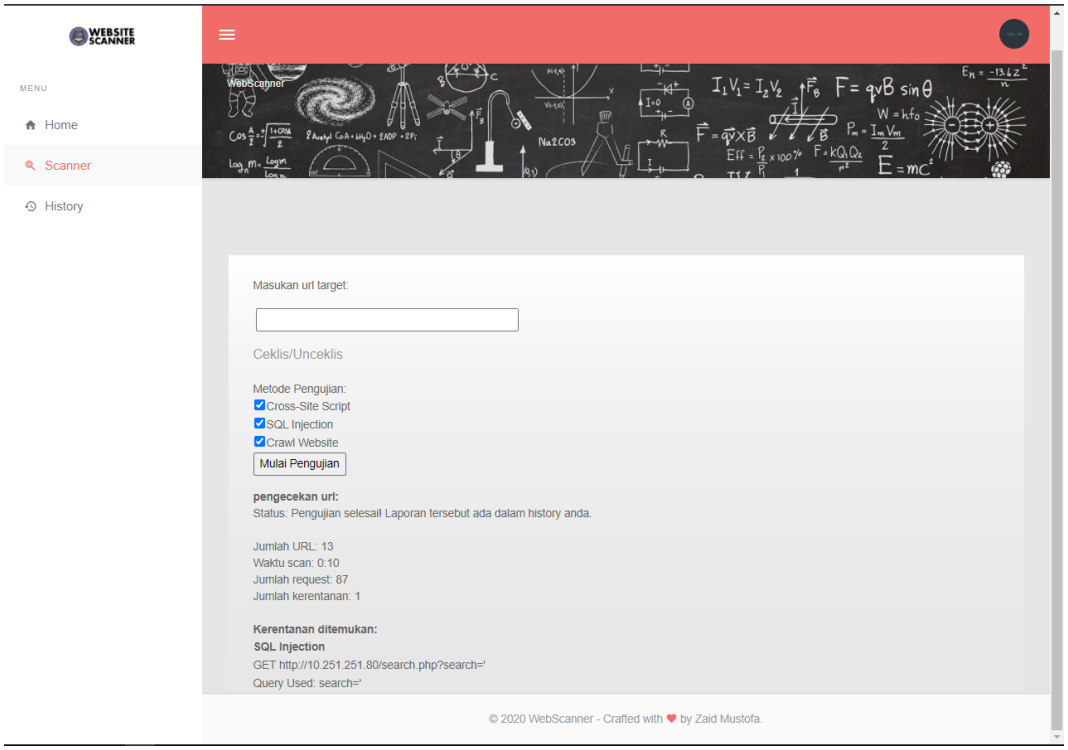
URL: http://testphp.vulnweb.com/listproducts.php?cat=<script>alert("hack by zaid mustofa");</script>
Method: GET
Query Used: cat=<script>alert("hack by zaid mustofa");</script>

URL: http://testphp.vulnweb.com//secured/newuser.php
Method: POST
Query Used: uuname=<script>alert("hack by zaid mustofa");</script>&upass=Abc123&upass2=Abc123&urname=Abc123&ucc=Abc123&uemail=Abc123&uphone=Abc123&signup=signup

Gambar 4.9 Report: Rentan terhadap *XSS* pada *website* testphp.vulnweb.com

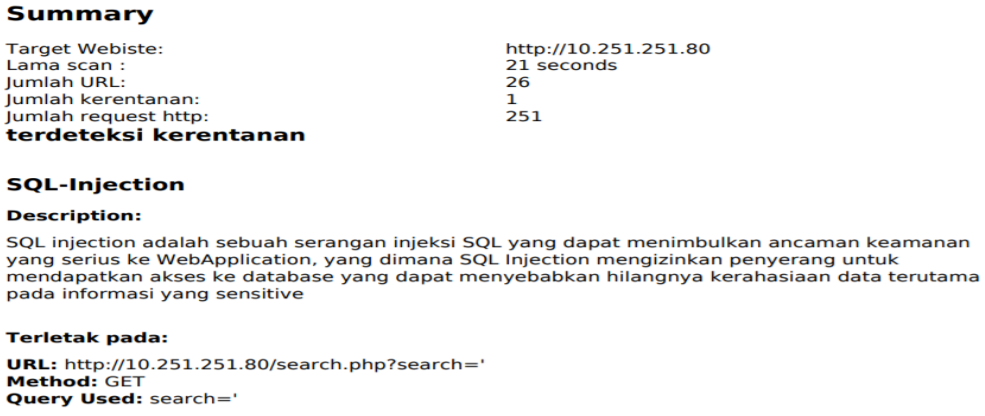
2. http://10.251.251.80

Pengujian ke dua terhadap *website* dari universitas ABC seperti gambar 4.10 :



Gambar 4.10 Proses scanner http://10.251.251.80

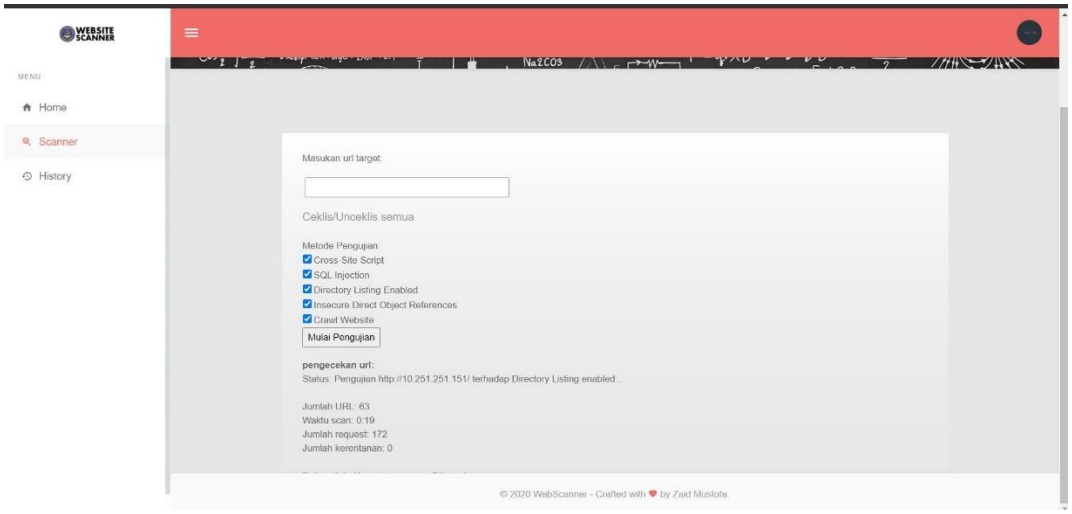
Pada gambar 4.11 menampilkan report scan dari *website* http://10.251.251.80 yang menunjukan pada url *search.php* mengindikasikan rentan terhadap *SQL Injction.*



Gambar 4.11 report scanner pada *website* http://10.251.251.80

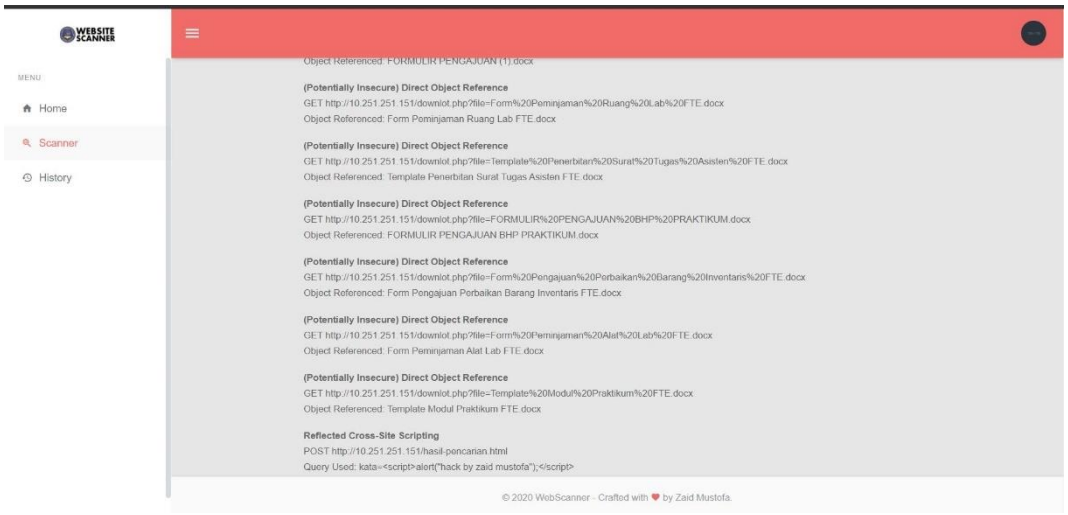
3. http://10.251.251.151

Pengujian ke tiga masih terhadap *website* dari universitas ABC seperti gambar 4.13 dengan url http://10.251.251.151 :



Gambar 4.12 Proses scanner http://10.251.251.151

Pada gambar 4.13 menampilkan bahwa aplikasi scanner mendeteksi bawah *website* http://10.251.251.151 memiliki kerentanan keamanan pada bagian *Insecure Cross-Site Script* :



Gambar 4.13 Kerentanan pada website http://10.251.251.151

4.2.3.2 Pengujian Secara Manual

Pada tahap ini akan diuji dengan membuka url yang terindikasi memliki kerentanan yang di keluarkan oleh aplikasi scanner. Tes manual akan dilakukan pada tiga alamat website http://tesphp.vulnweb.com, http://10.251.251.80, http://10.251.251.151 .

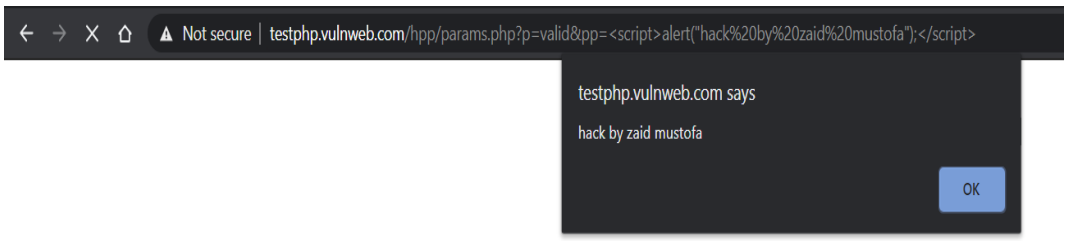
1. Pengujian manual http://testphp.vulnweb.com

Sebelumnya pada aplikasi scanner mendeteksi *website* http://tesphp.vulnweb.com rentan terhadap *SQL Injection*. Dan untuk memastikan kerentanan tersebut di test menggunakan *browser google chrome.*

Pada gambar 4.14 *website http://testphp.vulnweb.com* pada url *lisproducts.php* ketika di inputkan tanda petik satu, *website* merespon dengan menampilkan halaman *warning: mysql_fetch_array().*

Gambar 4.14 target http://testphp.vulnweb.com Rentan terhadap *SQL Injection*

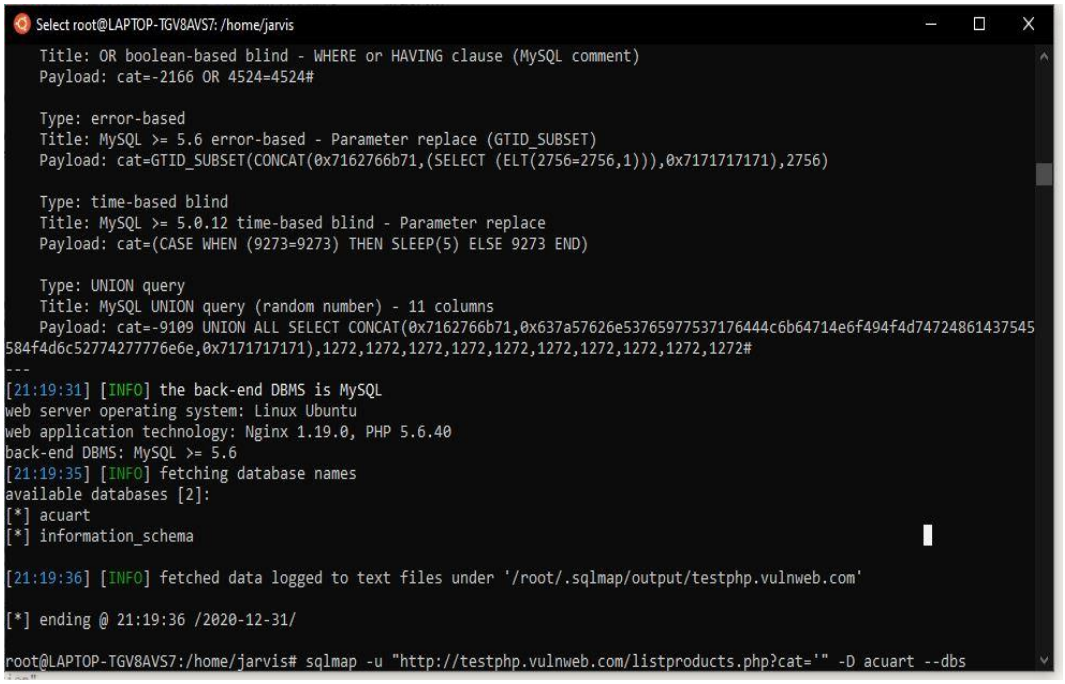
Selanjutnya Pada gambar 4.15 membuktikan bahwa *website* tersebut juga rentan terhadap *cross-site script,* diamana pada akhiran url tersebut disisipkan *script* <script>alert(“hack by zaid mustofa”);</script> dan *website* merespon *script* tadi dengan menampilkan pesan hack by zaid mustofa.



Gambar 4.15 target http://testphp.vulnweb.com Rentan terhadap *Cross-Site Script*

- *Sqlmap* http://testphp.vulnweb.com

  Hasil report dari aplikasi scanner terdapat url dari http://testphp.vulnweb.com yang rentan *SQL Injection.* Pada tools *SQL Map* dengan menginputkan script *"sqlmap -u "http://testphp.vulnweb.com/listproducts.php?cat="" --risk=3 --level=5 –dbs"* untuk mencoba masuk kedalam sistem database *website* tersebut seperti pada gambar 4.16 terlihat bahwa basis data yang digunakan adalah *Mysql, web server* menggunakan nginx, dengan menampilkan tabel dari basis data yang digunakan.

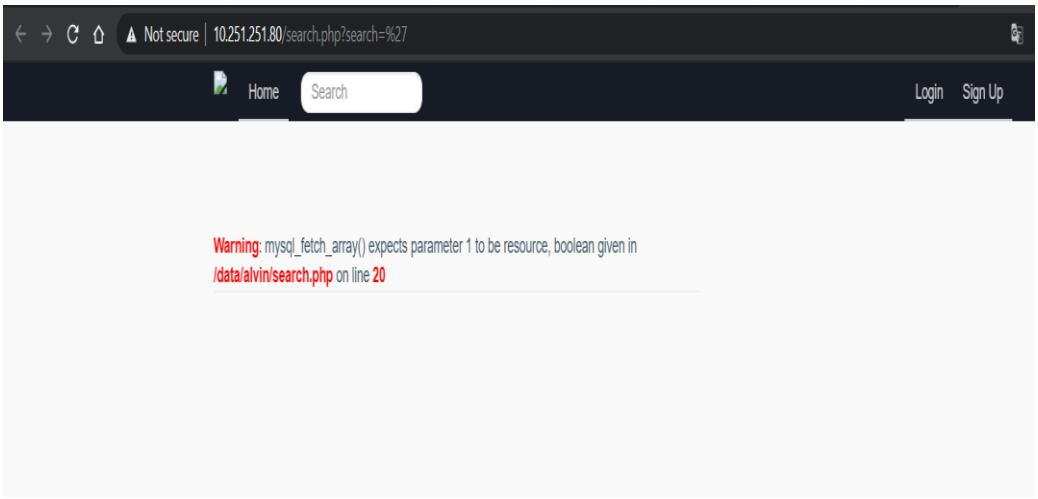


Gambar 4.16 *sqlmap* http://testphp.vulnweb.com

2. Pengujian manual http://10.251.251.80

Pengujian selanjutnya dilakukan pada alamat *website* http://10.251.251.80. Aplikasi Scanner menunjukkan bahwa situs tersebut rentan *SQL Injection*.

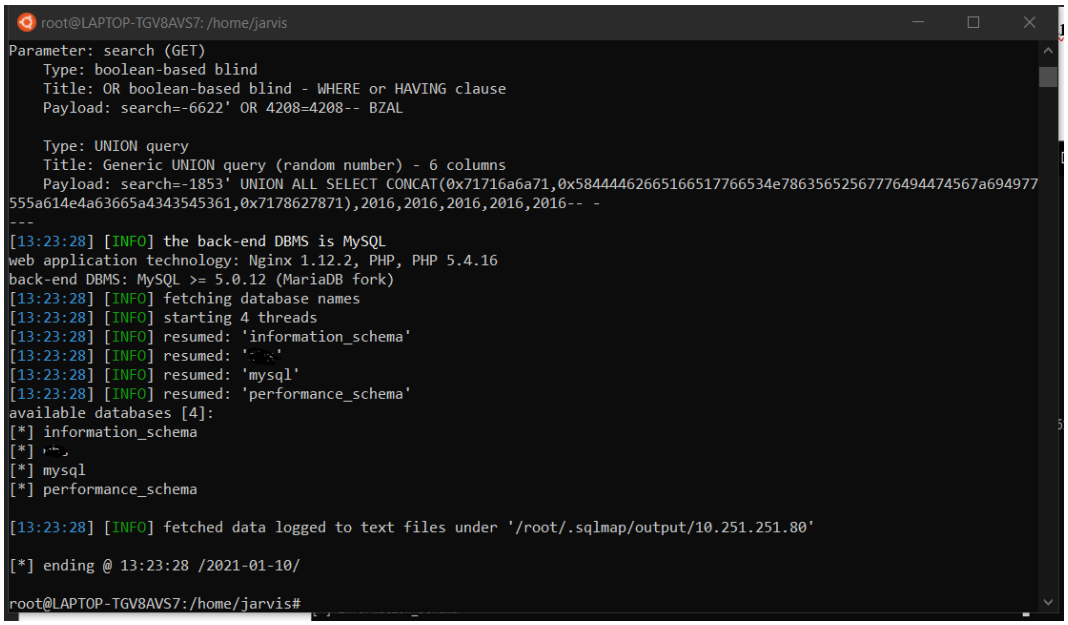
pada url *search.php* ketika di inputkan tanda petik satu, *website* merespon dengan menampilkan halaman *warning: mysql_fetch_array().* Dengan begitu *website* memang rentang *SQL Injection*



Gambar 4.17 target http://10.251.251.80 Rentan terhadap *SQL Injection*

- *Sqlmap* http://10.251.251.80

Dengan menggunakan tools *sqlmap* dan menginputkan script *"sqlmap -u " sqlmap -u "http://10.251.251.80/search.php?search=%27" --dbs"* untuk mencoba masuk kedalam sistem database *website* tersebut seperti pada gambar 4.18 terlihat bahwa basis data yang digunakan adalah *Mysql, web server* menggunakan nginx, dengan menampilkan tabel dari basis data yang digunakan.



Gambar 4.18 *sqlmap* http://10.251.251.80

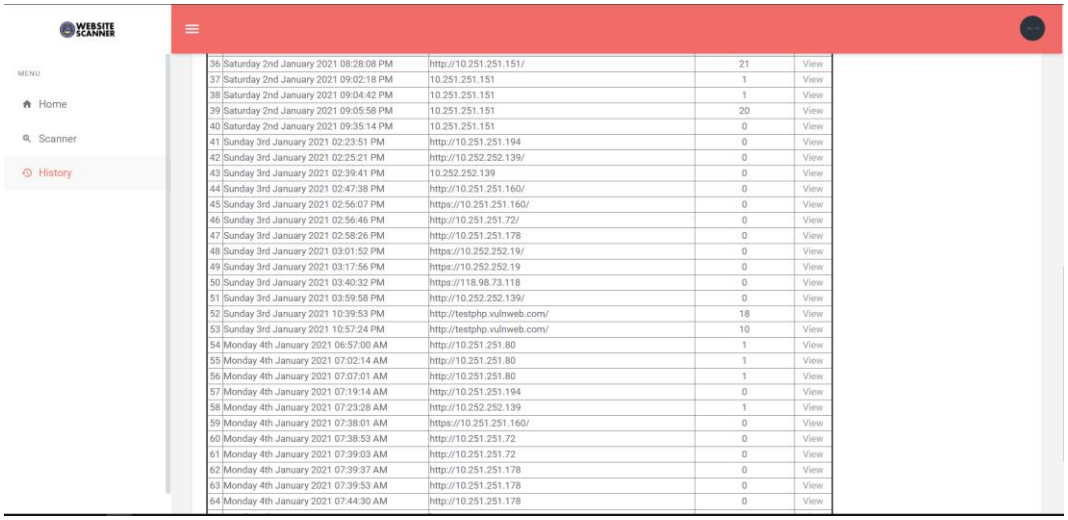
Untuk melihat hasil scanner seluruh target yang sudah ditentukan sebelumnya bisa di lihat pada table 4.6 dibawah ini :

Tabel 4.6 Hasil Scanner

No	Url Target	Sql Injection	Cross-Site Script
1	http://testphp.vulnweb.com	√	√
2	http://10.251.251.151	-	√
3	http://10.251.251.80	√	-
4	http://10.251.251.194	-	-
5	http://10.252.252.139	-	-
6	http://10.251.251.72	-	-
7	http://10.251.251.178	-	-
8	https://10.252.252.19	-	-

9	https://118.xx.xx.118	-	-

Menu History setelah melakukan semua pengujian scanner pada gambar 4.19 :



36	Saturday 2nd January 2021 08:28:08 PM	http://10.251.251.151/	21	View
37	Saturday 2nd January 2021 09:02:18 PM	10.251.251.151	1	View
38	Saturday 2nd January 2021 09:04:42 PM	10.251.251.151	1	View
39	Saturday 2nd January 2021 09:05:58 PM	10.251.251.151	20	View
40	Saturday 2nd January 2021 09:35:14 PM	10.251.251.151	0	View
41	Sunday 3rd January 2021 02:23:51 PM	http://10.251.251.194	0	View
42	Sunday 3rd January 2021 02:25:21 PM	http://10.252.252.139/	0	View
43	Sunday 3rd January 2021 02:39:41 PM	10.252.252.139	0	View
44	Sunday 3rd January 2021 02:47:38 PM	http://10.251.251.160/	0	View
45	Sunday 3rd January 2021 02:56:07 PM	https://10.251.251.160/	0	View
46	Sunday 3rd January 2021 02:56:46 PM	http://10.251.251.72/	0	View
47	Sunday 3rd January 2021 02:58:26 PM	http://10.251.251.178	0	View
48	Sunday 3rd January 2021 03:01:52 PM	https://10.252.252.19/	0	View
49	Sunday 3rd January 2021 03:17:56 PM	https://10.252.252.19	0	View
50	Sunday 3rd January 2021 03:40:32 PM	https://118.98.73.118	0	View
51	Sunday 3rd January 2021 03:59:58 PM	http://10.252.252.139/	0	View
52	Sunday 3rd January 2021 10:39:53 PM	http://testphp.vulnweb.com/	18	View
53	Sunday 3rd January 2021 10:57:24 PM	http://testphp.vulnweb.com/	10	View
54	Monday 4th January 2021 06:57:00 AM	http://10.251.251.80	1	View
55	Monday 4th January 2021 07:02:14 AM	http://10.251.251.80	1	View
56	Monday 4th January 2021 07:07:01 AM	http://10.251.251.80	1	View
57	Monday 4th January 2021 07:19:14 AM	http://10.251.251.194	0	View
58	Monday 4th January 2021 07:23:28 AM	http://10.252.252.139	1	View
59	Monday 4th January 2021 07:38:01 AM	https://10.251.251.160/	0	View
60	Monday 4th January 2021 07:38:53 AM	http://10.251.251.72	0	View
61	Monday 4th January 2021 07:39:03 AM	http://10.251.251.72	0	View
62	Monday 4th January 2021 07:39:37 AM	http://10.251.251.178	0	View
63	Monday 4th January 2021 07:39:53 AM	http://10.251.251.178	0	View
64	Monday 4th January 2021 07:44:30 AM	http://10.251.251.178	0	View

Gambar 4.19 History scanner dari seluruh target

BAB V

PENUTUP

5.1 Kesimpulan

a. Aplikasi scanner memberi kemudahan dalam melakukan pengujian *website*
b. Aplikasi scanner terbukti dapat mendeteksi celah keamanan *website* pada jenis serangan *SQL Injection* dan *Cross-Site Script.*
c. Aplikasi yang di bangun juga melengkapi pada penelitian yang dilakukan oleh (Angela, M. E. 2013) yaitu metode bukan hanya menguji *SQL Injection* saja dan (Yudha F. Panji A.M. 2018) pengecekan bukan berdasarkan *page* "*id*" tetapi mencoba merayapi semua url dengan mencari url yang ada *form* input html nya dan menginputkan *script SQL Injection dan Cross-site Script*.

5.2 Saran

a. Metode scan *SQL Injection* bisa di tambahkan lagi payload serangan nya sehingga varisi kode *SQL Injection* nya lebih banyak.
b. Metode scan *SQL Injection* belum terbukti bisa mendeteksi kerentanan pada *website* yang menggunakan basis data *MS SQL* ataupun *Oracle DB*
c. Pada menu login , fitur remember *Email* dan password belum bisa menyimpan sehingga setiap kali ingin login harus d ketik manual
d. Bug pada menu login, sehingga ketika sudah login dan mencoba mengakses menu login lagi masih di arahkan untuk memasukan email dan password.
e. Metode pengujian keamanan hanya menggunakan 2 metode yaitu *SQL Injection* dan *Cross-Site Script* sehingga masih diperlukan penambahan metode agar pengujian keamanan lebih baik

DAFTAR PUSTAKA

A.S Rosa dan Salahuddin M, 2018. Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek), INFORMATIKA , Bandung

Acunetix. 2020. *Why Is Directory Listing Dangerous?*. https://www.acunetix.com/blog/articles/directory-listing-information-disclosure/ diakses 16 desember 2020

Agung, Gregorius. 2000. Membuat Homepage Interaktif Dengan CGI/Perl. Jakarta: PT. Elex Media Koputindo.

Agus Saputra. 2012. Membuat Aplikasi Absensi dan Kuisioner untuk Panduan Skripsi. PT. Elex Media Koputindo. Jakarta

Andi Hamzah, 1993. Hukum Pidana yang berkaitan dengan komputer, Sina Grafika, jakarta.

BSSN. 2020. Rekap Serangan Siber (Januari – April 2020). https://www.kominfo.go.id/content/detail/3434/open-source-di-kominfo/0/program_prioritas#:~:text=Sumber%20terbuka%20(open%20source)%20adalah,biasanya%20menggunakan%20fasilitas%20komunikasi%20internet). diakses 10 desember 2020

Eddy Djunedi Karnasudiraja. 1993. Yurisprudensi Kejahatan Komputer, Jakarta, CV Tanjung Agung

Elu A, M. 2013. Rancang Bangun Aplikasi Pendeteksian *Vulnerability Structured Query Language (Sql) Injection* Untuk Keamanan Website

G Rodriguez, J Torres, E Benavides. 2019. *Cross-Site Scripting (XSS) Attacks And Mitigation: A Survey*

Justin Clarke. 2009. *SQL Injection Attack and Defense*. Burlington, MA 01803. Syngress Publishing, Inc., Elsevier, Inc.

Moh. Nazir. 1988. Metodologi Penelitian. Jakarta: Ghalia Indonesia.

Puspitosari, Heni A. Juli 2010. Pemrograman Web Database dengan PHP dan MySQL Tingkat Lanjut. Penerbit : Skripta. Malang.

Raharjo, Budi. 2015. Belajar Otodidak MySql. Bandung: Informatika.

Roger S. Pressman, 2002. Rekayasa Perangkat Lunak Pendekatan Praktisi (Buku Satu), ANDI Yogyakarta

Seth Fogie. 2007. *XSS Attacks*. Burlington, MA 01803. Syngress Publishing, Inc., Elsevier, Inc.

Varacode .*Application Security Vulnerability: Code Flaws, Insecure Code*. https://www.veracode.com/security/application-security-vulnerability-code-flaws-insecure-code. diakses tanggal 16 desember 2020

Varacode .*SQL Injection : Vulnerabilities & How To Prevent SQL Injection Attacks*. https://www.veracode.com/security/sql-injection. diakses tanggal 16 desember 2020

Widodo, 2011. Hukum Pidana di Bidang Teknologi Informasi CyberCrime Law :Telaah Teoritik dan Bedah Kasus, Yogyakarta:Aswaja Presindo.

Yudha, F. Panji A, M. 2018. Perancangan Aplikasi Pengujian Celah Keamanan Pada Aplikasi Berbasis Web

Yuhefizar, 1998. Desain Web dengan Microsoft FrontPage 97. Penerbit Wahana Komputer dan Andi : Semarang dan Yogyakarta.

Lampiran

Kode program

index.php

```
<?php
header("Location: view/login.php");
exit;
?>
```

dashboard.php

```
<?php
session_start();
require_once('scanner/functions/databaseFunctions.php');
require_once('session_control.php');
?>
<!DOCTYPE html>
<html lang="en">

    <head>
        <meta charset="utf-8" />
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width,
initial-scale=1.0, user-scalable=0, minimal-ui">
        <title>WebScanner</title>
        <meta content="Admin Dashboard" name="description" />
        <meta content="Themesbrand" name="zaid mustofa" />
        <link rel="shortcut icon"
href="assets/images/favicon.ico">

    </head>

    <body>

        <!-- Begin page -->
        <div id="wrapper">

            <!-- Top Bar Start -->
            <div class="topbar">

                <!-- LOGO -->
                <div class="topbar-left">
                    <a href="http://stmik-im.ac.id" class="logo">
                        <span>
                            <img src="assets/images/logo.png"
alt="" height="24">
                        </span>
                        <i>
                            <img src="assets/images/logo-sm.png"
alt="" height="22">
                        </i>
```

```
                    </a>
                </div>

                <nav class="navbar-custom">

                    <ul class="navbar-right d-flex list-inline
float-right mb-0">
                        <li class="dropdown notification-list d-
none d-sm-block">

                        </li>

                        <li class="dropdown notification-list">
                            <a class="nav-link dropdown-toggle
arrow-none waves-effect waves-light" data-toggle="dropdown"
href="#" role="button" aria-haspopup="false" aria-
expanded="false">

                            <div class="dropdown-menu dropdown-
menu-right dropdown-menu-lg">
                                <!-- item-->
                                <h6 class="dropdown-item-text">
                                    Notifications (37)
                                </h6>
                                <div class="slimscroll
notification-item-list">
                                    <!-- item-->
                                    <a href="javascript:void(0);"
class="dropdown-item notify-item active">
                                        <div class="notify-icon
bg-success"><i class="mdi mdi-cart-outline"></i></div>
                                        <p class="notify-
details">Your order is placed<span class="text-muted">Dummy text
of the printing and typesetting industry.</span></p>
                                    </a>
                                    <!-- item-->
                                    <a href="javascript:void(0);"
class="dropdown-item notify-item">
                                        <div class="notify-icon
bg-warning"><i class="mdi mdi-message"></i></div>
                                        <p class="notify-
details">New Message received<span class="text-muted">You have 87
unread messages</span></p>
                                    </a>
                                    <!-- item-->
                                    <a href="javascript:void(0);"
class="dropdown-item notify-item">
                                        <div class="notify-icon
bg-info"><i class="mdi mdi-flag"></i></div>
                                        <p class="notify-
details">Your item is shipped<span class="text-muted">It is a long
established fact that a reader will</span></p>
                                    </a>
                                    <!-- item-->
                                    <a href="javascript:void(0);"
class="dropdown-item notify-item">
```

```
                                        <div class="notify-icon
bg-primary"><i class="mdi mdi-cart-outline"></i></div>
                                        <p class="notify-
details">Your order is placed<span class="text-muted">Dummy text
of the printing and typesetting industry.</span></p>
                                    </a>
                                    <!-- item-->
                                    <a href="javascript:void(0);"
class="dropdown-item notify-item">
                                        <div class="notify-icon
bg-danger"><i class="mdi mdi-message"></i></div>
                                        <p class="notify-
details">New Message received<span class="text-muted">You have 87
unread messages</span></p>
                                    </a>
                                </div>
                                <!-- All-->
                                <a href="javascript:void(0);"
class="dropdown-item text-center text-primary">
                                    View all <i class="fi-arrow-
right"></i>
                                </a>
                            </div>
                        </li>
                        <li class="dropdown notification-list">
                            <div class="dropdown notification-list
nav-pro-img">
                                <a class="dropdown-toggle nav-link
arrow-none waves-effect nav-user waves-light" data-
toggle="dropdown" href="#" role="button" aria-haspopup="false"
aria-expanded="false">
                                    <img
src="assets/images/users/user-4.jpg" alt="user" class="rounded-
circle">
                                </a>
                                <div class="dropdown-menu
dropdown-menu-right profile-dropdown ">
                                    <!-- item-->

                                    <div class="dropdown-
divider"></div>
                                    <a class="dropdown-item text-
danger" href="login.php"></i> Login</a>
                                    <a class="dropdown-item text-
danger" href="logout.php?action=logout"><i class="mdi mdi-power
text-danger"></i> Logout</a>
                                </div>
                            </div>
                        </li>

                    </ul>

                    <ul class="list-inline menu-left mb-0">
                        <li class="float-left">
                            <button class="button-menu-mobile
open-left waves-effect waves-light">
```

```
                                <i class="mdi mdi-menu"></i>
                            </button>
                        </li>
                    </ul>

                </nav>

            </div>
            <!-- Top Bar End -->

            <!-- ========== Left Sidebar Start ========== -->
            <div class="left side-menu">
                <div class="slimscroll-menu" id="remove-scroll">

                    <!--- Sidemenu -->
                    <div id="sidebar-menu">
                        <!-- Left Menu Start -->
                        <ul class="metismenu" id="side-menu">
                            <li class="menu-title">Menu</li>
                            <li>
                                <a href="dashboard.php"
class="waves-effect">
                                    <i class="mdi mdi-
home"></i><span> Home </span>
                                </a>
                            </li>
                            <li>
                                <a href="scanner.php"
class="waves-effect">
                                    <i class="mdi mdi-search-
web"></i><span> Scanner </span>
                                </a>
                            </li>
                            <li>
                                <a href="history.php"
class="waves-effect">
                                    <i class="mdi mdi-
history"></i><span> History </span>
                                </a>
                            </li>
                        </ul>

                    </div>
                    <!-- Sidebar -->
                    <div class="clearfix"></div>

                </div>
                <!-- Sidebar -left -->

            </div>
            <!-- Left Sidebar End -->

            <!--
================================================================== -->
            <!-- Start right Content here -->
```

```
            <!--
================================================================ -->
            <div class="content-page">
                <!-- Start content -->
                <div class="content">
                    <div class="container-fluid">

                        <div class="row">
                            <div class="col-sm-12">
                                <div class="page-title-box">
                                    <h4 class="page-
title">Home</h4>
                                    <ol class="breadcrumb">
                                        <li class="breadcrumb-
item"><a href="javascript:void(0);">WebScanner</a></li>
                                    </ol>
                                </div>
                            </div>
                        </div>
                        <!-- end row -->

                        <div class="page-content-wrapper">
                            <div class="row">
                                <div class="col-12">
                                    <div class="card">
                                        <div class="card-body">
                                            <!-- Demo purpose only
-->
                                            <div style="min-
height: 300px;">
                                                <p>Selamat
datang</p>
                                            </div>
                                        </div>
                                    </div>
                                </div>
                            </div>
                        </div>
                        <!-- end page content-->

                    </div> <!-- container-fluid -->

                </div> <!-- content -->

                <footer class="footer">
                    © 2020 WebScanner <span class="d-none d-sm-
inline-block">- Crafted with <i class="mdi mdi-heart text-
danger"></i> by Zaid Mustofa.</span>
                </footer>

            </div>


            <!--
================================================================ -->
            <!-- End Right content here -->
```

```
            <!--
================================================================== -->


        </div>
        <!-- END wrapper -->


        <script src="../plugins/jquery-
sparkline/jquery.sparkline.min.js"></script>

        <!-- App js -->
        <script src="assets/js/app.js"></script>

    </body>

</html>
```

login.php

```
<?php
session_start();
require_once('scanner/functions/databaseFunctions.php');
require_once('session_control.php');
?>
<!DOCTYPE html>
<html lang="en">

    <head>
        <meta charset="utf-8" />
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width,
initial-scale=1.0, user-scalable=0, minimal-ui">
        <title>WebScanner</title>
        <meta content="Admin Dashboard" name="description" />
        <meta content="Themesbrand" name="zaid mustofa" />
        <link rel="shortcut icon"
href="assets/images/favicon.ico">
    </head>

    <body>

        <!-- Background -->
        <div class="account-pages"></div>
        <!-- Begin page -->
        <div class="wrapper-page">

            <div class="card">
                <div class="card-body">

                    <h3 class="text-center m-0">
                        <a href="index.html" class="logo logo-
admin"><img src="assets/images/logo.png" height="30"
alt="logo"></a>
```

```
                    </h3>

                    <div class="p-3">
                        <h4 class="text-muted font-18 m-b-5 text-
center">Welcome Back !</h4>
                        <p class="text-muted text-center"><?php
if(isset($loginMsg)){echo $loginMsg;}else{ echo "Sign in to
continue to WebScanner.";} ?></p>

                        <form class="form-horizontal m-t-30"
action="login.php" method="post">

                            <div class="form-group">
                                <label
for="username">Email</label>
                                <input name="email" class="form-
control" type="text" value="Email"
onfocus="if(this.value=='Email') this.value='';"
onblur="if(this.value=='') this.value='Email';"/>
                            </div>

                            <div class="form-group">
                                <label
for="userpassword">Password</label>
                                <input name="password"
class="form-control" type="text" autocomplete="off"
value="Password" onfocus="if(this.value=='Password'){
this.value='';this.type='password'}" onblur="if(this.value=='')
this.value='Password';"/>
                            </div>

                            <div class="form-group row m-t-20">
                                <div class="col-6">
                                    <div class="custom-control
custom-checkbox">
                                        <input type="checkbox"
class="custom-control-input" id="customControlInline">
                                        <label class="custom-
control-label" for="customControlInline">Remember me</label>
                                    </div>
                                </div>
                                <div class="col-6 text-right">
                                    <button class="btn btn-primary
w-md waves-effect waves-light" type="submit">Log In</button>
                                </div>
                            </div>

                            <div class="form-group m-t-10 mb-0
row">
                                <div class="col-12 m-t-20">
                                    <a href="register.php"
class="text-muted"><i class="mdi mdi-lock"></i> Don't have an
account ?</a>
                                </div>
                            </div>
                        </form>
```

```
                    </div>

                </div>
            </div>

        </div>

        <!-- END wrapper -->


        <script src="plugins/jquery-
sparkline/jquery.sparkline.min.js"></script>

        <!-- App js -->
        <script src="assets/js/app.js"></script>

    </body>

</html>
```

Logout.php

```
<?php

session_start();
$currentDir = './';
require_once($currentDir .
'scanner/functions/databaseFunctions.php');
ini_set('error_reporting', E_ALL);
?>
<!DOCTYPE html>
<head>
        <meta charset="utf-8" />
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width,
initial-scale=1.0, user-scalable=0, minimal-ui">
        <title>WebScanner</title>
        <meta content="Admin Dashboard" name="description" />
        <meta content="Themesbrand" name="author" />
        <link rel="shortcut icon"
href="assets/images/favicon.ico">

    </head>
<body>
<!--Header Begin-->
<div id="header">
  <div class="center">
    <div id="logo"><a href="#">WebVulScan</a></div>
    <!--Menu Begin-->
    <div id="menu">
    <?php
    require_once($currentDir . 'session_control.php');
    ?>
<body>
```

```
        <!-- Begin page -->
        <div id="wrapper">

            <!-- Top Bar Start -->
            <div class="topbar">

                <!-- LOGO -->
                <div class="topbar-left">
                    <a href="index.html" class="logo">
                        <span>
                            <img src="assets/images/logo.png"
alt="" height="24">
                        </span>
                        <i>
                            <img src="assets/images/logo-sm.png"
alt="" height="22">
                        </i>
                    </a>
                </div>

                <nav class="navbar-custom">

                    <ul class="navbar-right d-flex list-inline
float-right mb-0">
                        <li class="dropdown notification-list d-
none d-sm-block">
                            <form role="search" class="app-
search">

                            </form>
                        </li>


                            <div class="dropdown-menu dropdown-
menu-right dropdown-menu-lg">
                                <!-- item-->
                                <h6 class="dropdown-item-text">
                                    Notifications (37)
                                </h6>
                                <div class="slimscroll
notification-item-list">
                                    <!-- item-->
                                    <a href="javascript:void(0);"
class="dropdown-item notify-item active">
                                        <div class="notify-icon
bg-success"><i class="mdi mdi-cart-outline"></i></div>
                                        <p class="notify-
details">Your order is placed<span class="text-muted">Dummy text
of the printing and typesetting industry.</span></p>
                                    </a>
                                    <!-- item-->
                                    <a href="javascript:void(0);"
class="dropdown-item notify-item">
                                        <div class="notify-icon
bg-warning"><i class="mdi mdi-message"></i></div>
```

```
                                        <p class="notify-
details">New Message received<span class="text-muted">You have 87
unread messages</span></p>
                                    </a>
                                    <!-- item-->
                                    <a href="javascript:void(0);"
class="dropdown-item notify-item">
                                        <div class="notify-icon
bg-info"><i class="mdi mdi-flag"></i></div>
                                        <p class="notify-
details">Your item is shipped<span class="text-muted">It is a long
established fact that a reader will</span></p>
                                    </a>
                                    <!-- item-->
                                    <a href="javascript:void(0);"
class="dropdown-item notify-item">
                                        <div class="notify-icon
bg-primary"><i class="mdi mdi-cart-outline"></i></div>
                                        <p class="notify-
details">Your order is placed<span class="text-muted">Dummy text
of the printing and typesetting industry.</span></p>
                                    </a>
                                    <!-- item-->
                                    <a href="javascript:void(0);"
class="dropdown-item notify-item">
                                        <div class="notify-icon
bg-danger"><i class="mdi mdi-message"></i></div>
                                        <p class="notify-
details">New Message received<span class="text-muted">You have 87
unread messages</span></p>
                                    </a>
                                </div>
                                <!-- All-->
                                <a href="javascript:void(0);"
class="dropdown-item text-center text-primary">
                                    View all <i class="fi-arrow-
right"></i>
                                </a>
                            </div>
                        </li>
                        <li class="dropdown notification-list">
                            <div class="dropdown notification-list
nav-pro-img">
                                <a class="dropdown-toggle nav-link
arrow-none waves-effect nav-user waves-light" data-
toggle="dropdown" href="#" role="button" aria-haspopup="false"
aria-expanded="false">
                                    <img
src="assets/images/users/user-4.jpg" alt="user" class="rounded-
circle">
                                </a>
                                <div class="dropdown-menu
dropdown-menu-right profile-dropdown ">
                                    <!-- item-->
```

```
                                    <div class="dropdown-
divider"></div>
                                    <a class="dropdown-item text-
danger" href="login.php"></i> Login</a>
                                </div>
                            </div>
                        </li>

                    </ul>

                    <ul class="list-inline menu-left mb-0">
                        <li class="float-left">
                            <button class="button-menu-mobile
open-left waves-effect waves-light">
                                <i class="mdi mdi-menu"></i>
                            </button>
                        </li>
                    </ul>

                </nav>

            </div>
            <!-- Top Bar End -->

            <!-- ========== Left Sidebar Start ========== -->
            <div class="left side-menu">
                <div class="slimscroll-menu" id="remove-scroll">

                    <!--- Sidemenu -->
                    <div id="sidebar-menu">
                        <!-- Left Menu Start -->
                        <ul class="metismenu" id="side-menu">
                            <li class="menu-title">Menu</li>
                            <li>
                                <a href="dashboard.php"
class="waves-effect">
                                    <i class="mdi mdi-
home"></i><span> Home </span>
                                </a>
                            </li>
                            <li>
                                <a href="scanner.php"
class="waves-effect">
                                    <i class="mdi mdi-search-
web"></i><span> Scanner </span>
                                </a>
                            </li>
                            <li>
                                <a href="history.php"
class="waves-effect">
                                    <i class="mdi mdi-
history"></i><span> History </span>
                                </a>
                            </li>
                        </ul>
```

```
                    </div>
                    <!-- Sidebar -->
                    <div class="clearfix"></div>

                </div>
                <!-- Sidebar -left -->

            </div>
            <!-- Left Sidebar End -->

            <!--
================================================================== -->
            <!-- Start right Content here -->
            <!--
================================================================== -->
            <div class="content-page">
                <!-- Start content -->
                <div class="content">
                    <div class="container-fluid">

                        <div class="row">
                            <div class="col-sm-12">
                                <div class="page-title-box">
                                    <h4 class="page-
title">Home</h4>
                                    <ol class="breadcrumb">
                                        <li class="breadcrumb-
item"><a href="javascript:void(0);">WebScanner</a></li>
                                    </ol>
                                </div>
                            </div>
                        </div>
                        <!-- end row -->

                        <div class="page-content-wrapper">
                            <div class="row">
                                <div class="col-12">
                                    <div class="card">
                                        <div class="card-body">
                                            <!-- Demo purpose only
-->
                                            <div style="min-
height: 300px;">
                                                <p><?php
if(isset($loginMsg)){echo $loginMsg;}else{ echo "Sign in to
continue to WebScanner.";} ?></p>
                                            </div>

                        <!-- end page content-->

                    </div> <!-- container-fluid -->

                </div> <!-- content -->

                <footer class="footer">
```

```
                    © 2020 WebScanner <span class="d-none d-sm-
inline-block">- Crafted with <i class="mdi mdi-heart text-
danger"></i> by Zaid Mustofa.</span>
                </footer>

            </div>


            <!--
================================================================= -->
            <!-- End Right content here -->
            <!--
================================================================= -->


        </div>
        <!-- END wrapper -->

        <script src="../plugins/jquery-
sparkline/jquery.sparkline.min.js"></script>

        <!-- App js -->
        <script src="assets/js/app.js"></script>

    </body>

</html>
```

Register.php

```
<?php
session_start();
require_once('scanner/functions/databaseFunctions.php');
require_once('session_control.php');
?>
<!DOCTYPE html>
<html lang="en">

    <head>
        <meta charset="utf-8" />
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width,
initial-scale=1.0, user-scalable=0, minimal-ui">
        <title>WebScanner</title>
        <meta content="Admin Dashboard" name="description" />
        <meta content="Themesbrand" name="zaid mustofa" />
        <link rel="shortcut icon"
href="assets/images/favicon.ico">
    </head>

    <body>

        <!-- Background -->
        <div class="account-pages"></div>

        <!-- Begin page -->
```

```
        <div class="wrapper-page">
            <div class="card">
                <div class="card-body">

                    <h3 class="text-center m-0">
                        <a href="index.html" class="logo logo-
admin"><img src="assets/images/logo.png" height="30"
alt="logo"></a>
                    </h3>

                    <div class="p-3">
                        <h4 class="text-muted font-18 m-b-5 text-
center">Free Register</h4>
                        <p class="text-muted text-center"><?php
require_once('display_register_form.php'); ?></p>

                        <form class="form-horizontal m-t-30"
action="register.php" method="post">

                            <div class="form-group">
                                <label
for="useremail">Email</label>
                                <input name="email" class="form-
control" type="text" value="Email"
onfocus="if(this.value=='Email') this.value='';"
onblur="if(this.value=='') this.value='Email';"/>
                            </div>

                            <div class="form-group">
                                <label
for="username">Username</label>
                                <input name="regusername"
class="form-control" type="text" value="Username"
onfocus="if(this.value=='Username') this.value='';"
onblur="if(this.value=='') this.value='Username';"/>
                            </div>

                            <div class="form-group">
                                <label
for="userpassword">Password</label>
                                <input name="regpassword"
class="form-control" type="text" autocomplete="off"
value="Password" onfocus="if(this.value=='Password'){
this.value='';this.type='password';}" onblur="if(this.value=='')
this.value='Password';"/>
                            </div>

                            <div class="form-group">
                                <label for="userpassword">Confirm
Password</label>
                                <input name="regpassword2"
class="form-control" type="text" autocomplete="off" value="Confirm
Password" onfocus="if(this.value=='Confirm Password'){
this.value=''; this.type='password';}" onblur="if(this.value=='')
this.value='Confirm Password';"/>
                            </div>
```

```
                            <div class="form-group row m-t-20">
                                <div class="col-12 text-right">
                                    <button class="btn btn-primary
w-md waves-effect waves-light" type="submit">Register</button>
                                </div>
                            </div>

                            <div class="form-group m-t-10 mb-0
row">
                                <div class="col-12 m-t-20">
                                    <p class="font-14 text-muted
mb-0">Already have an account ? <a href="login.php" class="text-
primary">Login</a></p>
                                </div>


        <!-- END wrapper -->


        <script src="plugins/jquery-
sparkline/jquery.sparkline.min.js"></script>

        <!-- App js -->
        <script src="assets/js/app.js"></script>

    </body>

</html>
```

Scanner.php

```
<?php
session_start();
$currentDir = './';
require_once($currentDir .
'scanner/functions/databaseFunctions.php');
?>
<!DOCTYPE html>
<html lang="en">

    <head>
        <meta charset="utf-8" />
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width,
initial-scale=1.0, user-scalable=0, minimal-ui">
        <title>WebScanner</title>
        <meta content="Admin Dashboard" name="description" />
        <meta content="Themesbrand" name="zaid mustofa" />
        <link rel="shortcut icon"
href="assets/images/favicon.ico">
        <meta charset="windows-1252">
<link rel="shortcut icon" href="images/favicon.gif" />
<link rel="stylesheet" type="text/css" href="style.css" />
```

```
<script type="text/javascript" src="jquery-1.6.4.js"></script>
    </head>

    <body>

        <!-- Begin page -->
        <div id="wrapper">

            <!-- Top Bar Start -->
            <div class="topbar">

                <!-- LOGO -->
                <div class="topbar-left">
                    <a href="index.html" class="logo">
                        <span>
                            <img src="assets/images/logo.png"
alt="" height="24">
                        </span>
                        <i>
                            <img src="assets/images/logo-sm.png"
alt="" height="22">
                        </i>
                    </a>
                </div>

                <nav class="navbar-custom">

                    <ul class="navbar-right d-flex list-inline
float-right mb-0">
                        <li class="dropdown notification-list d-
none d-sm-block">
                        </li>


                            <div class="dropdown-menu dropdown-
menu-right dropdown-menu-lg">
                                <!-- item-->
                                <h6 class="dropdown-item-text">
                                    Notifications (37)
                                </h6>
                                <div class="slimscroll
notification-item-list">
                                    <!-- item-->
                                    <a href="javascript:void(0);"
class="dropdown-item notify-item active">
                                        <div class="notify-icon
bg-success"><i class="mdi mdi-cart-outline"></i></div>
                                        <p class="notify-
details">Your order is placed<span class="text-muted">Dummy text
of the printing and typesetting industry.</span></p>
                                    </a>
                                    <!-- item-->
                                    <a href="javascript:void(0);"
class="dropdown-item notify-item">
                                        <div class="notify-icon
bg-warning"><i class="mdi mdi-message"></i></div>
```

```
                                        <p class="notify-
details">New Message received<span class="text-muted">You have 87
unread messages</span></p>
                                    </a>
                                    <!-- item-->
                                    <a href="javascript:void(0);"
class="dropdown-item notify-item">
                                        <div class="notify-icon
bg-info"><i class="mdi mdi-flag"></i></div>
                                        <p class="notify-
details">Your item is shipped<span class="text-muted">It is a long
established fact that a reader will</span></p>
                                    </a>
                                    <!-- item-->
                                    <a href="javascript:void(0);"
class="dropdown-item notify-item">
                                        <div class="notify-icon
bg-primary"><i class="mdi mdi-cart-outline"></i></div>
                                        <p class="notify-
details">Your order is placed<span class="text-muted">Dummy text
of the printing and typesetting industry.</span></p>
                                    </a>
                                    <!-- item-->
                                    <a href="javascript:void(0);"
class="dropdown-item notify-item">
                                        <div class="notify-icon
bg-danger"><i class="mdi mdi-message"></i></div>
                                        <p class="notify-
details">New Message received<span class="text-muted">You have 87
unread messages</span></p>
                                    </a>
                                </div>
                                <!-- All-->
                                <a href="javascript:void(0);"
class="dropdown-item text-center text-primary">
                                    View all <i class="fi-arrow-
right"></i>
                                </a>
                            </div>
                        </li>
                        <li class="dropdown notification-list">
                            <div class="dropdown notification-list
nav-pro-img">
                                <a class="dropdown-toggle nav-link
arrow-none waves-effect nav-user waves-light" data-
toggle="dropdown" href="#" role="button" aria-haspopup="false"
aria-expanded="false">
                                    <img
src="assets/images/users/user-4.jpg" alt="user" class="rounded-
circle">
                                </a>
                                <div class="dropdown-menu
dropdown-menu-right profile-dropdown ">
                                    <!-- item-->
```

```
                                    <div class="dropdown-
divider"></div>
                                    <a class="dropdown-item text-
danger" href="login.php"></i> Login</a>
                                    <a class="dropdown-item text-
danger" href="logout.php?action=logout"><i class="mdi mdi-power
text-danger"></i> Logout</a>
                                </div>
                            </div>
                        </li>

                    </ul>

                    <ul class="list-inline menu-left mb-0">
                        <li class="float-left">
                            <button class="button-menu-mobile
open-left waves-effect waves-light">
                                <i class="mdi mdi-menu"></i>
                            </button>
                        </li>
                    </ul>

                </nav>

            </div>

            <!-- Top Bar End -->

            <!-- ========== Left Sidebar Start ========== -->
            <div class="left side-menu">
                <div class="slimscroll-menu" id="remove-scroll">

                    <!--- Sidemenu -->
                    <div id="sidebar-menu">
                        <!-- Left Menu Start -->
                        <ul class="metismenu" id="side-menu">
                            <li class="menu-title">Menu</li>
                            <li>
                                <a href="dashboard.php"
class="waves-effect">
                                    <i class="mdi mdi-
home"></i><span> Home </span>
                                </a>
                            </li>
                            <li>
                                <a href="scanner.php"
class="waves-effect">
                                    <i class="mdi mdi-search-
web"></i><span> Scanner </span>
                                </a>
                            </li>
                            <li>
                                <a href="history.php"
class="waves-effect">
                                    <i class="mdi mdi-
history"></i><span> History </span>
```

```
                                </a>
                            </li>
                        </ul>

                    </div>
                    <!-- Sidebar -->
                    <div class="clearfix"></div>

                </div>
                <!-- Sidebar -left -->

            </div>
            <!-- Left Sidebar End -->

            <!--
================================================================ -->
            <!-- Start right Content here -->
            <!--
================================================================ -->
 <div class="content-page">
                <!-- Start content -->
                <div class="content">
                    <div class="container-fluid">

                        <div class="row">
                            <div class="col-sm-12">
                                <div class="page-title-box">
                                    <h4 class="page-
title">Scanner</h4>
                                    <ol class="breadcrumb">
                                        <li class="breadcrumb-
item"><a href="javascript:void(0);">WebScanner</a></li>
                                    </ol>
                                </div>
                        <!-- end row -->


<body>
    <div id="menu">
    <?php require_once($currentDir . 'session_control.php'); ?>
    </div>
    <!--Menu END-->
  </div>
</div>
<!--Header END-->

<!--SubPage MiddleRow Begin-->
<div id="midrow">
  <div class="center">
    <div class="textbox2">
      <p><?php require_once($currentDir .
'scanner/scanner_form.php'); ?></p>
    </div>
  </div>
</div>
<!--MiddleRow END-->
```

```
</body>
                    </div> <!-- container-fluid -->

                </div> <!-- content -->

                <footer class="footer">
                    © 2020 WebScanner <span class="d-none d-sm-
inline-block">- Crafted with <i class="mdi mdi-heart text-
danger"></i> by Zaid Mustofa.</span>
                </footer>

            </div>


            <!--
================================================================== -->
            <!-- End Right content here -->
            <!--
================================================================== -->


        </div>
        <!-- END wrapper -->

        <script src="../plugins/jquery-
sparkline/jquery.sparkline.min.js"></script>

        <!-- App js -->
        <script src="assets/js/app.js"></script>

    </body>

</html>
```

History.php

```
<?php
session_start();
$currentDir = './';
require_once($currentDir .
'scanner/functions/databaseFunctions.php');
?>
<!DOCTYPE html>
<html lang="en">

    <head>
        <meta charset="utf-8" />
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width,
initial-scale=1.0, user-scalable=0, minimal-ui">
        <title>WebScanner</title>
        <meta content="Admin Dashboard" name="description" />
        <meta content="Themesbrand" name="author" />
```

```
        <link rel="shortcut icon"
href="assets/images/favicon.ico">

type="text/css">
    </head>

    <body>

        <!-- Begin page -->
        <div id="wrapper">

            <!-- Top Bar Start -->
            <div class="topbar">

                <!-- LOGO -->
                <div class="topbar-left">
                    <a href="index.html" class="logo">
                        <span>
                            <img src="assets/images/logo.png"
alt="" height="24">
                        </span>
                        <i>
                            <img src="assets/images/logo-sm.png"
alt="" height="22">
                        </i>
                    </a>
                </div>


                <nav class="navbar-custom">

                    <ul class="navbar-right d-flex list-inline
float-right mb-0">
                        <li class="dropdown notification-list d-
none d-sm-block">
                        </li>

                            <div class="dropdown-menu dropdown-
menu-right dropdown-menu-lg">
                                <!-- item-->
                                <h6 class="dropdown-item-text">
                                    Notifications (37)
                                </h6>
                                <div class="slimscroll
notification-item-list">
                                    <!-- item-->
                                    <a href="javascript:void(0);"
class="dropdown-item notify-item active">
                                        <div class="notify-icon
bg-success"><i class="mdi mdi-cart-outline"></i></div>
                                        <p class="notify-
details">Your order is placed<span class="text-muted">Dummy text
of the printing and typesetting industry.</span></p>
                                    </a>
                                    <!-- item-->
```

```
                                    <a href="javascript:void(0);"
class="dropdown-item notify-item">
                                        <div class="notify-icon
bg-warning"><i class="mdi mdi-message"></i></div>
                                        <p class="notify-
details">New Message received<span class="text-muted">You have 87
unread messages</span></p>
                                    </a>
                                    <!-- item-->
                                    <a href="javascript:void(0);"
class="dropdown-item notify-item">
                                        <div class="notify-icon
bg-info"><i class="mdi mdi-flag"></i></div>
                                        <p class="notify-
details">Your item is shipped<span class="text-muted">It is a long
established fact that a reader will</span></p>
                                    </a>
                                    <!-- item-->
                                    <a href="javascript:void(0);"
class="dropdown-item notify-item">
                                        <div class="notify-icon
bg-primary"><i class="mdi mdi-cart-outline"></i></div>
                                        <p class="notify-
details">Your order is placed<span class="text-muted">Dummy text
of the printing and typesetting industry.</span></p>
                                    </a>
                                    <!-- item-->
                                    <a href="javascript:void(0);"
class="dropdown-item notify-item">
                                        <div class="notify-icon
bg-danger"><i class="mdi mdi-message"></i></div>
                                        <p class="notify-
details">New Message received<span class="text-muted">You have 87
unread messages</span></p>
                                    </a>
                                </div>
                                <!-- All-->
                                <a href="javascript:void(0);"
class="dropdown-item text-center text-primary">
                                    View all <i class="fi-arrow-
right"></i>
                                </a>
                            </div>
                        </li>
                        <li class="dropdown notification-list">
                            <div class="dropdown notification-list
nav-pro-img">
                                <a class="dropdown-toggle nav-link
arrow-none waves-effect nav-user waves-light" data-
toggle="dropdown" href="#" role="button" aria-haspopup="false"
aria-expanded="false">
                                    <img
src="assets/images/users/user-4.jpg" alt="user" class="rounded-
circle">
                                </a>
```

```
                                    <div class="dropdown-menu
dropdown-menu-right profile-dropdown ">
                                        <!-- item-->

                                        <div class="dropdown-
divider"></div>
                                        <a class="dropdown-item text-
danger" href="login.php"></i> Login</a>
                                        <a class="dropdown-item text-
danger" href="logout.php?action=logout"><i class="mdi mdi-power
text-danger"></i> Logout</a>

                                    </div>
                                </div>
                            </li>

                        </ul>

                        <ul class="list-inline menu-left mb-0">
                            <li class="float-left">
                                <button class="button-menu-mobile
open-left waves-effect waves-light">
                                    <i class="mdi mdi-menu"></i>
                                </button>
                            </li>
                        </ul>

                    </nav>

                </div>
                <!-- Top Bar End -->

                <!-- ========== Left Sidebar Start ========== -->
                <div class="left side-menu">
                    <div class="slimscroll-menu" id="remove-scroll">

                        <!--- Sidemenu -->
                        <div id="sidebar-menu">
                            <!-- Left Menu Start -->
                            <ul class="metismenu" id="side-menu">
                                <li class="menu-title">Menu</li>
                                <li>
                                    <a href="dashboard.php"
class="waves-effect">
                                        <i class="mdi mdi-
home"></i><span> Home </span>
                                    </a>
                                </li>
                                <li>
                                    <a href="scanner.php"
class="waves-effect">
                                        <i class="mdi mdi-search-
web"></i><span> Scanner </span>
                                    </a>
                                </li>
                                <li>
```

```
                                <a href="history.php"
class="waves-effect">
                                    <i class="mdi mdi-
history"></i><span> History </span>
                                </a>
                            </li>
                        </ul>

                    </div>
                    <!-- Sidebar -->
                    <div class="clearfix"></div>

                </div>
                <!-- Sidebar -left -->

            </div>
            <!-- Left Sidebar End -->

            <!--
================================================================== -->
            <!-- Start right Content here -->
            <!--
================================================================== -->
            <div class="content-page">
                <!-- Start content -->
                <div class="content">
                    <div class="container-fluid">

                        <div class="row">
                            <div class="col-sm-12">
                                <div class="page-title-box">
                                    <h4 class="page-
title">History</h4>
                                    <ol class="breadcrumb">
                                        <li class="breadcrumb-
item"><a href="javascript:void(0);">WebScanner</a></li>
                                    </ol>
                                </div>
                            </div>
                        </div>
                        <!-- end row -->

                        <div class="page-content-wrapper">
                            <div class="row">
                                <div class="col-12">
                                    <div class="card">
                                        <div class="card-body">
                                            <!-- Demo purpose only
-->
                                            <div style="min-
height: 300px;">
                                                <p><?php
require_once($currentDir . 'scanner/scan_history.php');?></p>
                                            </div>
                                        </div>
                                    </div>
```

```
                                </div>
                            </div>
                        </div>
                        <!-- end page content-->

                    </div> <!-- container-fluid -->

                </div> <!-- content -->

                <footer class="footer">
                    © 2020 WebScanner <span class="d-none d-sm-
inline-block">- Crafted with <i class="mdi mdi-heart text-
danger"></i> by Zaid Mustofa.</span>
                </footer>

            </div>


            <!--
================================================================== -->
            <!-- End Right content here -->
            <!--
================================================================== -->


        </div>
        <!-- END wrapper -->

        <script src="../plugins/jquery-
sparkline/jquery.sparkline.min.js"></script>

        <!-- App js -->
        <script src="assets/js/app.js"></script>

    </body>

</html>
```

Session_control.php

```
<?php

if(isset($_GET['action']))
{
    $action = $_GET['action'];
    if($action == 'logout')
    {
        session_start();
        unset($_SESSION['username']);
        $loginMsg = 'Anda berhasil logout';
        echo $loginMsg;

    }
}
```

```
if(isset($_POST['email']) && isset($_POST['password']))
{
    $email = $_POST['email'];
    $password = $_POST['password'];

    $continueLogin = true;

    if(!filter_var($email, FILTER_VALIDATE_EMAIL) ||
!ctype_alnum($password))
    {
        $loginMsg = 'kesalahan Email atau Password. Silahkan
periksa kembali';
        $continueLogin = false;
    }

    if(connectToDb($db) && $continueLogin)
    {
        $query = "SELECT * FROM users WHERE email = '$email' AND
password = SHA1('$password')";
        $result = $db->query($query);
        if($result)
        {
            $numRows = $result->num_rows;
            if($numRows == 0)
                $loginMsg = 'kesalahan Email atau Password.
Silahkan periksa kembali';
            else
            {
                $row = $result->fetch_object();
                $username = $row->username;
                $_SESSION['username'] = $username;
                $_SESSION['email'] = $email;
                $loginMsg = 'Selamt anda berhasil login';
                header("Location: dashboard.php");
            }
        }
        else
        {
            $loginMsg = 'There was a problem checking your
credentials. Please contact administrator if the problem
persists';
        }
    }
}
```

Display_register_form.php

```
<?php

if(isset($_SESSION['username']))
{
    echo 'Anda harus keluar untuk membuat akun';
}
else
```

```
{
    if(isset($_POST['regusername']) ||
isset($_POST['regpassword']) || isset($_POST['regpassword2']) ||
             isset($_POST['email']))
    {
        if(empty($_POST['regusername']) ||
empty($_POST['regpassword']) || empty($_POST['regpassword2']) ||
             empty($_POST['email']))
        {
            echo 'Anda harus mengisi semua kolom masukan';
        }
        else if($_POST['regpassword'] != $_POST['regpassword2'])
        {
            echo 'Konfirmasi kata sandi tidak cocok dengan kata
sandi pertama yang dimasukkan';
        }
        else if(!ctype_alnum($_POST['regusername']) ||
!ctype_alnum($_POST['regpassword']))//only hav to check the first
password as the second password entered is equal to this (checked
above)
        {
            echo 'Nama pengguna dan sandi harus alfanumerik.
Silahkan coba lagi';
        }
        else if(!filter_var($_POST['email'],
FILTER_VALIDATE_EMAIL))
        {
            echo 'Alamat email yang dimasukkan tampaknya bukan
email yang valid. Jika ini adalah alamat email yang valid, harap
hubungi administrator kami';
        }
        else//everything should be ok if we make it to here
        {
            $username = $_POST['regusername'];
            $password = $_POST['regpassword'];
            $email = $_POST['email'];

            if(connectToDb($db))
            {
                $query = "SELECT * FROM users WHERE username =
'$username'";
                $result = $db->query($query);
                if($result)
                {
                    $numRows = $result->num_rows;
                    if($numRows > 0)
                        echo 'punten, username ini sudah
terdaftar, silahkan menggunakan yang lain';
                    else
                    {
                        $query = "SELECT * FROM users WHERE email
= '$email'";
                        $result = $db->query($query);
                        if($result)
                        {
                            $numRows = $result->num_rows;
```

```
                            if($numRows > 0)
                                echo 'punten, Email ini sudah
terdaftar, silahkan menggunakan yang lain';
                            else
                            {
                                $query = "INSERT INTO users
VALUE('$username',SHA1('$password'),'$email')";
                                $result = $db->query($query);
                                if($result)
                                {
                                    echo 'Selamat! Anda berhasil
mendaftar, Silahkan login untuk menggunakan aplikasi';
                                }
                                else
                                    echo 'Ada masalah saat
menghubungkan ke database. Silakan hubungi administrator jika
masalah terus berlanjut';
                            }
                        }
                        else
                            echo 'Ada masalah saat menghubungkan
ke database. Silakan hubungi administrator jika masalah terus
berlanjut';
                    }
                }
                else
                    echo 'Ada masalah saat menghubungkan ke
database. Silakan hubungi administrator jika masalah terus
berlanjut';

            }
            else
                echo 'Ada masalah saat menghubungkan ke database.
Silakan hubungi administrator jika masalah terus berlanjut';

        }
    }
}
?>
```

Scanner_form.php

```
<?Php

?>

<script type="text/javascript">
Feature beginscan(fee,valuetwo,valuethree,valuefour,valuefive)
    jquery.Publish("scanner/begin_scan.Personal home page",
specifiedurl:value, testid:valuetwo, username:valuethree,
electronic mail:valuefour,testcases:valuefive);


Function sizetbl(h)
```

```
  var tbl = report.Getelementbyid('tbl');
  tbl.Style.Show = h;


Checked=true;
Function checkedall (form1)

    var aa = report.Getelementbyid('form1');
    if (checked == genuine)

        checked = fake

    else

        checked = proper

    for (var i =0; i < aa.Elements.Duration; i++)

     aa.Elements[i].Checked = checked;


</script>

<?Php

Require_once('functions/databaseFunctions.Php');
Require_once('classes/Logger.Php');

If(isset($_SESSION['username']))

    $username = $_SESSION['username'];

    if(isset($_SESSION['email']))
        $email = $_SESSION['email'];
    else
        $email = ''; //maybe email to administrator
?>

    <body>
    <form id="form1" name="form1" method="post" >
        <p>masukan url goal:</p>
        <p>
          <label for="urlToScan"></label>
          <input type="text" size="40" name="urlToScan"
id="urlToScan" />
        <br>

        <div id=tbl name=tbl style="overflow:visible">
        <a href="javascript:checkedAll(form1)"><font
size="3">ceklis/unceklis semua</font></a><br>
        <br>metode pengujian:<br>
        <table border="0">

        <tr><td><input type="checkbox" name="rxss" value="rxss"
checked /></td><td>cross-web site script</td></tr>
```

```
        <tr><td><input type="checkbox" name="sqli" value="sqli"
checked /></td><td>sq. Injection</td></tr>
        <tr><td><input type="checkbox" name="crawlurl"
value="crawlurl" checked /></td><td>move slowly internet
site</td></tr>
        </table>
        </div>
        <p>
          <input type="submit" class="button" name="submit"
id="submit" value="Mulai Pengujian" />
        </p>
     </form>

<?Php

    if(isset($_POST['urlToScan']))

        $testCases = ' ';//options
        if(isset($_POST['rxss'])) $testCases .= $_POST['rxss'] . '
';
        if(isset($_POST['sqli'])) $testCases .= $_POST['sqli'] . '
';

        if(isset($_POST['crawlurl'])) $testCases .=
$_POST['crawlurl'] . ' ';

        $urlToScan = trim($_POST['urlToScan']);
        if(!Empty($urlToScan))

            $log = new Logger();
            $log->lfile('scanner/logs/eventlogs');

            $log->lwrite('connecting to database');

            $connectionflag = connecttodb($db);

            if(!$connectionflag)

                $log->lwrite('error tidak terbuhung ke database');
                echo 'errors tidak terbuhung ke database';
                return;


            $log->lwrite('generating subsequent test id');
            $nextid = generatenexttestid($db);

            if(!$nextid)

                $log->lwrite(identifikasi selanjutnya yang
dihasilkan adalah nol ');
                echo 'identifikasi selanjutnya yang dihasilkan
adalah nol;
                return;

            else
```

```
                $log->lwrite("next identity generated is
$nextid");
                $testid = $nextid;
                $now = time();
                $question = "insert into
tests(identification,fame,numurlsfound,kind,num_requests_sent,star
t_timestamp,finish_timestamp,scan_finished,url,username,urls_found
) values($nextid,'growing profile for brand new
test...',zero,'experiment',zero,$now,$now,zero,'$urltoscan','$user
name','')";
                $end result = $db->question($question);
                if(!$result)

                    $log->lwrite("hassle executing query:
$question ");
                    echo 'trouble placing a brand new take a look
at into the database. Please strive again.';
                    go back;

                else

                    $log->lwrite("correctly done question: $query
");



            updatestatus($db, 'pending...', $testid);

            $query = "replace exams set numurlsfound = zero where
identification = $testid;";
            $db->query($query);
            $question = "update assessments set period = zero in
which identification = $testid;";
            $db->query($question);

                echo '<script type="text/javascript">
                $(record).Geared up(characteristic()
                 $.Put up("scanner/getstatus.Php", testid:' .
"$testid" . ',
characteristic(information)$("#reputation").Html(information));
               var refreshid = setinterval(characteristic()
                  $.Put up("scanner/getstatus.Hypertext
Preprocessor", testid:' . "$testid" . ',
feature(information)$("#popularity").Html(facts));
               , 500);
               $.Ajaxsetup( cache: false );
                );</script>';

                echo '<script type="text/javascript">
                $(report).Equipped(characteristic()
                 $.Post("scanner/getvulnerabilities.Hypertext
Preprocessor", testid:' . "$testid" . ',
characteristic(statistics)$("#scanstatus").Html(statistics));
               var refreshid = setinterval(feature()
```

```
                  $.Post("scanner/getvulnerabilities.Hypertext
Preprocessor", testid:' . "$testid" . ',
function(information)$("#scanstatus").Html(records));
               , a thousand);
               $.Ajaxsetup( cache: fake );
                );</script>';

                $urltoscan = $_post['urlToScan'];

                $log->lwrite('calling ajax function
begincrawl()');
                echo '<script type="text/javascript">';
                echo
"beginscan('$urltoscan','$testid','$username','$e mail',
'$testcases');";
                echo '</script>';


        else
            echo 'error: tidak ada url yang dimasukan';


    echo '<div id="status"></div><br>';
    echo '<div id="scanstatus"></div><br>';

Else
    echo 'sekarang anda belum login. Mohon untuk login terlebih
dahulu.';
?>
```

Scanner_history.php

```
<?Php

Require_once('functions/databaseFunctions.Php');

Global $user;

If(isset($_SESSION['username']))

    $username = $_SESSION['username'];

    if(!ConnectToDb($db))

        echo 'There was a problem connecting to the database';
        return;


    $query = "SELECT * FROM tests WHERE type = 'scan' AND username
= '$username'";
    //echo $query;
    $result = $db->question($question);
    if($result)
```

```
        $numrows = $end result->num_rows;
        if($numrows == zero)
            echo 'anda belum pernah melakukan pemindaian
sebelumnya';
        else

            echo '<table border="3"
width="900"><tr><th>identity</th><th>waktu
start</th><th>url</th><th>jumlah
kerentanan</th><th>file</th></tr>';
            for($i=zero; $i<$numRows; $i++)

                $row = $result->fetch_object();
                $id = $row->identification;
                $starttime = $row->start_timestamp;
                $starttimeformatted = date('l js f y h:i:s a',
$starttime);
                $url = $row->url;

                $numvulns = 'unknown';
                $query = "select * from test_results in which
test_id = $identity";
                $resulttwo = $db->query($query);
                if($resulttwo)
                    $numvulns = $resulttwo->num_rows;

                $report = '<a href="scanner/reports/Test_' . $id .
'.Pdf" target="_blank">view</a>';

                echo '<tr>';
                echo "<td align='center'>$identification</td>";
                echo "<td align='left'>$starttimeformatted</td>";
                echo "<td align='left'>$url</td>";
                echo "<td align='center'>$numvulns</td>";
                echo "<td align='center'>$record</td>";
                echo '</tr>';


            echo '</table>';




    else
        echo 'there was a hassle retrieving your facts from the
database';

Else
    echo 'sekrang anda belum login. Mohon untuk login terlebih
dahulu';


?>
```

Begin_scan.php

```
<?Php

Set_time_limit(0);
Error_reporting(E_ALL);

$currentDir = './';


Require_once($currentDir .
"../crawler/PHPCrawl_071/classes/phpcrawler.Class.Php");
Require_once($currentDir .
"../crawler/PHPCrawl_071/classes/mycrawler.Php");


Require_once($currentDir .
'classes/simplehtmldom/simple_html_dom.Php');
Require_once($currentDir . 'classes/httpclient-2011-08-
21/http.Php');


Require_once($currentDir . 'classes/Form.Php');
Require_once($currentDir . 'classes/InputField.Php');
Require_once($currentDir . 'classes/Logger.Php');
Require_once($currentDir . 'classes/PostOrGetObject.Php');
Require_once($currentDir . 'classes/Vulnerability.Php');


Require_once($currentDir . 'functions/commonFunctions.Php');
Require_once($currentDir . 'functions/databaseFunctions.Php');
Require_once($currentDir . 'functions/createPdfReport.Php');

Require_once($currentDir . 'tests/testForReflectedXSS.Php');
Require_once($currentDir . 'tests/testForSQLi.Php');

Require_once($currentDir . 'classes/tcpdf/config/lang/eng.Php');
Require_once($currentDir . 'classes/tcpdf/tcpdf.Php');

$log = new Logger();
$log->lfile($currentdir . 'logs/eventlogs');

$log->lwrite('menghubungkan ke database');

$connectionflag = connecttodb($db);

Isset($_post['specifiedUrl']) ? $urltoscan =
$_post['specifiedUrl'] : $urltoscan = '';
Isset($_post['testId']) ? $testid = $_post['testId'] : $testid =
0;
Isset($_post['username']) ? $username = $_post['username'] :
$username = 'user';
Isset($_post['email']) ? $email = $_post['email'] : $e mail =
'diazzaid20@gmail.Com';//admin deal with
```

```
Isset($_post['testCases']) ? $testcases = $_post['testCases'] :
$testcases = '';//admin cope with

If(empty($urltoscan))

    echo 'tidak ada url experiment';
    $log->lfile('tidak ada url test');
    return;


If(stripos($urltoscan, 'http') !== 0)
    $urltoscan = 'http://' . $urltoscan;

$log->lwrite("url untuk di test: $urltoscan");

$query = "replace exams set repute = 'preparing crawl for
$urltoscan' in which id = $testid;";
$db->question($question);

//check if crawling is enabled
$crawlurlflag = false;
If(stristr($testcases,' crawlurl ') !== fake)
    $crawlurlflag = real;

If($crawlurlflag)

    $log->lwrite('instantiating crawler');
    $crawler = &new mycrawler();
    $crawler->seturl($urltoscan);
    $crawler->settestid($testid);
    $crawler->addreceivecontenttype("/textpngproper);
    $crawler->setfirstcrawl(actual);
    $crawler->settestid($testid);


    updatestatus($db, "crawling $urltoscan...", $testid);
    $log->lwrite('memulai crawler');

    $crawler->pass();
    $urlsfound = $crawler->urlsfound;

Else
    $urlsfound = array($urltoscan);

$logstr = sizeof($urlsfound) . ' url ditemukan untuk pengujian: '
. $testid;

$log->lwrite("semua url ditemukan tidak termasuk pengecualian:");
Foreach($urlsfound as $currenturl)
    $log->lwrite($currenturl);

$sitebeingtested = getsitebeingtested($urltoscan);

If(stristr($testcases,' dirlist ') !== false)

    //test area for listing listing enabled
```

```
    $log->lwrite("memulai $urltoscan pengujian directory list
enabled");
    testdirectorylistingenabled($urlsfound[0], $sitebeingtested,
$testid, $crawlurlflag); //the primary url in the array is usually
the overall domain name e.G. Http://www.Abc.Com
    $log->lwrite("selesai $urltoscan pengujian directory listing
enabled: $testid");
    updatestatus($db, "selesai $urltoscan pengujian listing list
enabled...", $testid);


If(stristr($testcases,' idor ') !== false)

    //test all urls for insecure direct item references
    $log->lwrite('memulai pengujian insecure direct object
references');
    testdirectobjectrefs($urlsfound, $testid);
    $log->lwrite('pengujian insecure direct object references
selesai: ' . $testid);
    updatestatus($db, "pengujian insecure direct object references
selesai...", $testid);


If(stristr($testcases,' rxss ') !== false)

    //check all urls for pondered pass-site scripting
    $log->lwrite('memulai pengujian setiap url untuk pondered
xss');
    for($i=0; $i<sizeof($urlsFound); $i++)

        testForReflectedXSS($urlsFound[$i], $siteBeingTested,
$testId);

    $log->lwrite('pengujian reflected xss selesai: ' . $testid);
    updatestatus($db, "pengujian reflected xss selesai...",
$testid);


If(stristr($testcases,' sqli ') !== fake)

    //take a look at all urls for sq. Injection
    $log->lwrite('memulai pengujian setiap url untuk sq.
Injection');
    for($i=zero; $i<sizeof($urlsFound); $i++)

        testForSQLi($urlsFound[$i], $siteBeingTested, $testId);

    $log->lwrite('pengujian sq. Injection selesai: ' . $testid);
    updatestatus($db, "pengujian square injection selesai...",
$testid);



//create pdf document
$log->lwrite('mulai membuat laporan pdf: ' . $testid);
Createpdfreport($testid, $filename);
```

```
$log->lwrite('selesai membuat laporan pdf: ' . $testid);
Updatestatus($db, "selesai membuat laporan pdf...", $testid);


$query = "update exams set scan_finished = 1 in which
identification = $testid;";
$end result = $db->query($question);

If(stristr($testcases,' emailpdf ') !== fake)
    updatestatus($db, "pengujian selesai! Laporan tersebut telah
dikirim ke e mail anda dan juga ada dalam riwayat pemindaian
anda.", $testid);
Else
    updatestatus($db, "pengujian selesai! Laporan tersebut ada
dalam history anda.", $testid);

$db->close();
?>
```

getstatus.php

```
<?Php

$currentDir = './';
Require_once($currentDir . 'functions/databaseFunctions.Php');
//require_once('classes/Logger.Php');

Isset($_POST['testId']) ? $testId = $_POST['testId'] : $testId =
0;

ConnectToDb($db);

$query = "SELECT * FROM tests WHERE id = $testId;";
$result = $db->query($query);
$row = $end result->fetch_object();
$completed = $row->scan_finished;

//update finish time to cutting-edge time even as test is not
finished
If($completed == 0)

    $now = time();
    $query = "update checks set finish_timestamp = $now where id =
$testid;";
    $result = $db->question($query);


$question = "choose * from exams wherein id = $testid;";
$result = $db->question($question);

$row = $end result->fetch_object();
$status = $row->fame;
$starttime = $row->start_timestamp;
$fintime = $row->finish_timestamp;
```

```
$depend = $row->numurlsfound;
$numrequests = $row->num_requests_sent;

$duration = $fintime - $starttime;
$mins = intval($period/60);
$seconds = $period % 60;
$secondsstr = strval($seconds);
$secondsformatted = str_pad($secondsstr,2,"zero",str_pad_left);

$question = "choose * from test_results where test_id = $testid;";
$end result = $db->query($query);
$numvulns = $end result->num_rows;

//todo: put table here, seems bit messy
Echo '<b>pengecekan url:</b><br>';
Echo 'reputation: ' . $repute;

Echo "<br><br>jumlah url: $matter";
Echo "<br>waktu test: $minutes:$secondsformatted";
Echo "<br>jumlah request: $numrequests";
Echo "<br>jumlah kerentanan: $numvulns";

$result->free();
$db->close();

?>
```

Getvulnerabilities.php

```
<?Php

$currentDir = './';
Require_once($currentDir . 'functions/databaseFunctions.Php');

Isset($_POST['testId']) ? $testId = $_POST['testId'] : $testId =
0;

$query = 'SELECT * FROM test_results WHERE test_id = ' . $testId;
ConnectToDb($db);
$result = $db->question($query);
If($end result)

    $numrows = $result->num_rows;
    if($numrows > zero)

        echo '<b>kerentanan ditemukan:</b>';

        for($i=zero; $i<$numRows; $i++)

            $row = $result->fetch_object();
            $type = $row->kind;
            $approach = strtoupper($row->approach);
            $url = $row->url;
            $data = $row->attack_str;
```

```
            if($type == 'rxss')

                $type = 'reflected cross-web page scripting';
                $information = 'query used: ' . $info;

            else if($kind == 'sqli')

                $kind = 'square injection';
                $info = 'question used: ' . $data;


            echo "<p><b>$type</b><br>";
            $urlhtml = htmlspecialchars($url);
            echo "$technique $urlhtml<br>";
            $infohtml = htmlspecialchars($info);
            echo "$infohtml</p>";

        $result->free();
        $db->close();

    else

        echo '<b>belum ada kerentanan yang ditemukan</b>';


?>
```

Testforsqli.php

```
<?Php

Set_time_limit(0);

Function testForSQLi($urlToCheck, $urlOfSite, $testId)

ConnectToDb($db);
UpdateStatus($db, "Pengujian $urlToCheck terhadap SQL
Injection...", $testId);

$log = new Logger();
$log->lfile('logs/eventlogs');

$log->lwrite("memulai pengujian square injection $urltocheck");

$posturl = $urltocheck;

$posturlpath = parse_url($posturl, php_url_path);

//periksa url tidak merespons dengan kode 5xx
$log->lwrite("memeriksa dari mana kode respon diterima
$urltocheck");
$https = new http_class;
$https->timeout=0;
$https->data_timeout=0;
```

```
$https->user_agent="mozilla/five.0 (windows nt 6.1; wow64)
applewebkit/537.21 (khtml, like gecko) chrome/forty
one.Zero.2228.Zero safari/537.21";
$https->follow_redirect=1;
$https->redirection_limit=5;
$https->settestid($testid);

$error=$http->getrequestarguments($urltocheck,$arguments);

$blunders=$http->open($arguments);

$log->lwrite("url yang akan diminta adalah $urltocheck");

If($blunders=="")

    $log->lwrite("mengirim permintaan http ke $urltocheck");
    $blunders=$http->sendrequest($arguments);

    if($errors=="")

        $headers=array();
        $blunders=$http->readreplyheaders($headers);
        if($mistakes=="")

            $responsecode = $http->response_status;//that is a
string
            $log->lwrite("menerima kode tanggapan:
$responsecode");
            if(intval($responsecode) >= 500 &&
intval($responsecode) <600)

                $log->lwrite("kode respon: $responsecode received
from: $urltocheck");
                return;



    $http->near();

If(strlen($error))

    echo "<H2 align="center">error: ",$mistakes,"</H2>n";
    $log->lwrite("blunders: $blunders");


$html = file_get_html($posturl, $testid);

If(empty($html))


    updatestatus($db, "problem getting contents from
$urltocheck...", $testid);
    $log->lwrite("masalah mendapatkan konten dari $urltocheck");
    return;
```

```
$log->lwrite("berhasil mendapat konten dari $urltocheck");

//kode serangan sq.
$arrayofpayloads = array( "'",
                          '"',
                          ';',
                          ')',
                          '(',
                          '.',
                          '--');


$arrayofsqlwarnings = array(
        "furnished argument isn't a legitimate mysql", //mysql
        "mysql_fetch_array\(\)",
        "on mysql result index",
        "you have got an errors to your sq. Syntax;",
        "you have an blunders on your sq. Syntax near",
        "mysql server version for the right syntax to apply",
        "\[MySQL\]\[odbc",
        "column depend would not fit",
        "the used pick out statements have one-of-a-kind variety
of columns",
        "desk '[^']+' doesn't exist",
        "db blunders: unknown blunders",
        ":[s]*mysql",
        "mysql_fetch",
        "machine\.Records\.Oledb\.Oledbexception", //ms sq.
        "\[SQL Server\]",
        "\[Microsoft\]\[ODBC SQL Server Driver\]",
        "\[SQLServer JDBC Driver\]",
        "\[sqlexception",
        "gadget.Records.Sqlclient.Sqlexception",
        "unclosed citation mark after the person string",
        "'80040e14'",
        "mssql_query\(\)",
        "odbc_exec\(\)",
        "microsoft ole db company for odbc drivers",
        "microsoft ole db provider for sq. Server",
        "wrong syntax near",
        "syntax error in string in query expression",
        "adodb\.Field \(0x800a0bcd\)<br>",
        "procedure '[^']+' requires parameter '[^']+'",
        "adodb\.Recordset'",
        "microsoft sq. Native patron errors",
        "unclosed citation mark after the individual string",
        "sqlcode", //db2"
        "db2 square blunders:",
        "sqlstate",
        "sybase message:", //sybase
        "syntax errors in question expression", //get entry to
        "information kind mismatch in standardsora)-[0-9][0-9][0-
9][0-9]", //oracle
        "postgresql question failed:", //postgre
        "supplied argument is not a legitimate postgresql result",
        "pg_query\(\) \[:",
```

```
        "pg_exec\(\) \[:",
        "com\.Informix\.Jdbc", //informix
        "dynamic page technology errors:",
        "dynamic square blunders",
        "\[DM_QUERY_E_SYNTAX\]", //dml
        "has happened in the location of:",
        "a parser blunders \(syntax mistakes\)",
        "java\.Sq.\.Sqlexception",//java
        "\[Macromedia\]\[SQLServer JDBC Driver\]" //coldfusion
        );


$parsedurl = parse_url($urltocheck);
$log->lwrite("periksa apakah $urltocheck berisi parameter");
If($parsedurl)

    if(isset($parsedurl['query']))

        $log->lwrite("$urltocheck memang mengandung parameter");

        $scheme = $parsedurl['scheme'];
        $host = $parsedurl['host'];
        $route = $parsedurl['path'];

        $query = $parsedurl['query'];
        parse_str($question,$parameters);
        $originalquery = $query;

        foreach($arrayofpayloads as $currentpayload)


            $https = new http_class;
            $https->timeout=zero;
            $https->data_timeout=0;

            $https->user_agent="mozilla/5.0 (home windows nt 6.1;
wow64) applewebkit/537.21 (khtml, like gecko)
chrome/41.Zero.2228.0 safari/537.21";
            $https->follow_redirect=1;
            $https->redirection_limit=5;
            $https->settestid($testid);

            foreach($parameters as $para)

                $question = $originalquery;

                $newquery = str_replace($para, $currentpayload,
$question);
                $query = $newquery;

                $testurl = $scheme . '://' . $host . $path . '?' .
$question;

                $log->lwrite("url yang akan diminta adalah:
$testurl");
```

```
                $mistakes=$https-
>getrequestarguments($testurl,$arguments);

                $mistakes=$http->open($arguments);
                if($errors=="")

                    $log->lwrite("mengirim permintaan http ke
$testurl");

                    $errors=$https->sendrequest($arguments);

                    if($mistakes=="")

                        $headers=array();
                        $errors=$http->readreplyheaders($headers);
                        if($blunders=="")

                            $errors = $https-
>readwholereplybody($body);

                            if(strlen($mistakes) == 0)

                                $vulnerabilityfound = fake;

                                for($warningindex=0; $warningindex
< sizeof($arrayOfSQLWarnings); $warningIndex++)

                                    $regularExpression =
"/$arrayOfSQLWarnings[$warningIndex]/";


if(preg_match($regularExpression,$body))

                                        $log->lwrite("menemukan
ekspresi reguler: $regularexpression, dalam isi respons http");
                                        $vulnerabilityfound =
proper;
                                        break;



                                if($vulnerabilityfound)


                                    echo '<br>square injection
gift!<br>question: ' . Htmlspecialchars($urltocheck) . '<br>';
                                    echo 'method: get <br>';
                                    echo 'url: ' .
Htmlspecialchars($testurl) . '<br>';
                                    echo 'errors: ' .
$regularexpression . '<br>';
                                    $tablename = 'test' . $testid;


                                    $square = "pick * from
test_results in which test_id = $testid and type = 'sqli' and
```

```
method = 'get' and url = '" . Addslashes($testurl) . "' and
attack_str = '" . Addslashes($question) . "'";
                                    $end result = $db-
>question($sq.);
                                    if(!$result)
                                        $log->lwrite("tidak dapat
menjalankan question $square");
                                    else

                                        $log->lwrite("query
berhasil dieksekusi $square");
                                        $numrows = $result-
>num_rows;
                                        if($numrows == 0)

                                            $log->lwrite("jumlah
baris adalah $numrows untuk question: $square");
                                            inserttestresult($db,
$testid, 'sqli', 'get', addslashes($testurl),
addslashes($question));


                                        $result->loose();

                                    $http->near();
                                    wreck 2;




                    $https->near();

                if(strlen($blunders))
                    echo "<H2 align="center">error:
",$mistakes,"</H2>n";




Else
    $log->lwrite("tidak dapat mengurai url yang rusak:
$urltocheck");


$arrayofforms = array();

$arrayofinputfields = array();

$log->lwrite("searching $posturl for forms");

$formnum = 1;
Foreach($html->find('form') as $shape)

    isset($form->attr['id']) ? $formid = htmlspecialchars($form-
>attr['id']) : $formid = '';
```

```
    isset($form->attr['name']) ? $formname =
htmlspecialchars($shape->attr['name']) : $formname ='';
    isset($shape->attr['method']) ? $formmethod =
htmlspecialchars($shape->attr['method']) : $formmethod = 'get';
    isset($shape->attr['action']) ? $formaction =
htmlspecialchars($form->attr['action']) : $formaction = '';

    $formmethod = strtolower($formmethod);


    if(empty($formaction))

        $strlengthurl = strlen($urltocheck);
        $strlengthsite = strlen($urlofsite);
        $firstindexofslash = strpos($urltocheck, '/',
$strlengthsite-1);
        $formaction = substr($urltocheck, $firstindexofslash+1,
$strlengthurl);


    $log->lwrite("ditemukan shape di $posturl: $formid $formname
$formmethod $formaction $formnum");

    $newform = new form($formid, $formname, $formmethod,
$formaction, $formnum);
    array_push($arrayofforms, $newform);

    foreach($form->find('enter') as $enter)

        isset($enter->attr['id']) ? $inputid =
htmlspecialchars($input->attr['id']) : $inputid = '';
        isset($enter->attr['name']) ? $inputname =
htmlspecialchars($enter->attr['name']) : $inputname = '';
        isset($enter->attr['value']) ? $inputvalue =
htmlspecialchars($input->attr['value']) : $inputvalue = '';
        isset($input->attr['type']) ? $inputtype =
htmlspecialchars($input->attr['type']) : $inputtype = '';

        $log->lwrite("found enter subject on $posturl: $inputid
$inputname $formid $formname $inputvalue $inputtype $formnum");

        $inputfield = new inputfield($inputid, $inputname,
$formid, $formname, $inputvalue, $inputtype, $formnum);

        array_push($arrayofinputfields, $inputfield);


    $formnum ++;



$log->lwrite('memulai pengujian bureaucracy');
For($i=0; $i<sizeof($arrayOfForms); $i++)

    $currentForm = $arrayOfForms[$i];
    $currentFormId = $currentForm->getid();
```

```
    $currentformname = $currentform->getname();
    $currentformmethod = $currentform->getmethod();
    $currentformaction = $currentform->getaction();
    $currentformnum = $currentform->getformnum();

    $arrayofcurrentformsinputs = array();

    $log->lwrite("memulai pengujian shape di $posturl:
$currentformid $currentformname $currentformmethod
$currentformaction");

    echo sizeof($arrayofinputfields) . "<br>";
    for($j=0; $j<sizeof($arrayOfInputFields); $j++)

        $currentInput = $arrayOfInputFields[$j];
        $currentInputIdOfForm = $currentInput->getidofform();
        $currentinputnameofform = $currentinput->getnameofform();
        $currentinputformnum = $currentinput->getformnum();


        if($currentformnum == $currentinputformnum)

            array_push($arrayofcurrentformsinputs, $currentinput);



    $log->lwrite("memulai pengujian input fields dari shape di
$posturl: $currentformid $currentformname $currentformmethod
$currentformaction");

    for($okay=zero; $k<sizeof($arrayofcurrentformsinputs); $ok++)

        echo sizeof($arrayofcurrentformsinputs) . '<br>';

        for($plindex=0; $plindex<sizeof($arrayOfPayloads);
$plIndex++)

            $currentFormInput = $arrayOfCurrentFormsInputs[$k];
            $currentFormInputName =  $currentFormInput->getname();
            $currentforminputtype =  $currentforminput->gettype();
            $currentforminputvalue =  $currentforminput-
>getvalue();

            if($currentforminputtype!= 'reset')

                $https = new http_class;
                $https->timeout=0;
                $https->data_timeout=zero;

                $http->user_agent="mozilla/five.0 (windows nt 6.1;
wow64) applewebkit/537.21 (khtml, like gecko) chrome/41.0.2228.0
safari/537.21";
                $https->follow_redirect=1;
                $https->redirection_limit=five;
                $https->settestid($testid);
```

```
                $defaultstr = 'abc123';

                $arrayofvalues = array();


                $otherinputs = array();

                for($l=0; $l<sizeof($arrayOfCurrentFormsInputs);
$l++)

                    if($currentFormInput->getname() !=
$arrayofcurrentformsinputs[$l]->getname())

                        array_push($otherinputs,
$arrayofcurrentformsinputs[$l]);



                $postobject = new
postorgetobject($currentforminputname,
$arrayofpayloads[$plIndex]);
                $log->lwrite("mengirim payload:
$arrayofpayloads[$plIndex], untuk enter area:
$currentforminputname");

                array_push($arrayofvalues, $postobject);

                for($m=zero; $m<sizeof($otherInputs); $m++)

                    $currentOther = $otherInputs[$m];
                    $currentOtherType = $currentOther->gettype();
                    $currentothername = $currentother->getname();
                    $currentothervalue = $currentother-
>getvalue();

                    if($currentothertype == 'text
$currentothertype == 'password')

                        $postobject = new
postorgetobject($currentothername, $defaultstr);
                        array_push($arrayofvalues, $postobject);
                    post')

                        $postobject  = new
postorgetobject($currentothername, $currentothervalue);
                        array_push($arrayofvalues, $postobject);

                    else if($currentothertype == 'radio')

                        $postobject  = new
postorgetobject($currentothername, $currentothervalue);

                        $discovered = false;
                        for($n = 0; $n < sizeof($arrayOfValues);
$n++)
```

```
                            if($arrayOfValues[$n]->getname() ==
$postobject->getname())

                                $located = authentic;
                                spoil;


                        if(!$found)
                            array_push($arrayofvalues,
$postobject);



                echo '<br><br>';

                if($currentformmethod == 'get')


                    if($urlofsite[strlen($urlOfSite)-1] == '/')
                        $actionurl = $urlofsite .
$currentformaction;
                    else
                        $actionurl = $urlofsite . '/' .
$currentformaction;

                    $totalteststr = '';
                    for($p=0; $p<sizeof($arrayOfValues); $p++)

                        $currentPostValue = $arrayOfValues[$p];
                        $currentPostValueName = $currentPostValue-
>getname();
                        $currentpostvaluevalue =
$currentpostvalue->getvalue();

                        $totalteststr .= $currentpostvaluename;
                        $totalteststr .= '=';
                        $totalteststr .= $currentpostvaluevalue;

                        if( $p != (sizeof($arrayofvalues) - 1) )
                            $totalteststr .= '&';



                    if(strpos($actionurl, '?')!==false)
                        $actionurl .= '&';
                    else
                        $actionurl .= '?';

                    $actionurl .= $totalteststr;

                    $errors=$http-
>getrequestarguments($actionurl,$arguments);

                    $errors=$http->open($arguments);
```

```
                    $log->lwrite("url to be asked is:
$actionurl");

                    if($errors=="")

                        $log->lwrite("sending http request to
$actionurl");
                        $mistakes=$http->sendrequest($arguments);

                        if($errors=="")

                            $headers=array();
                            $mistakes=$http-
>readreplyheaders($headers);
                            if($blunders=="")

                                $blunders = $http-
>readwholereplybody($frame);

                                if(strlen($errors) == 0)

                                    $vulnerabilityfound = fake;

                                    for($warningindex=zero;
$warningindex < sizeof($arrayOfSQLWarnings); $warningIndex++)

                                        $regularExpression =
"/$arrayOfSQLWarnings[$warningIndex]/";


if(preg_match($regularExpression,$body))

                                            $log-
>lwrite("discovered normal expression: $regularexpression, in
frame of http response");
                                            $vulnerabilityfound =
authentic;
                                            spoil;



                                    if($vulnerabilityfound)



                                        $totalteststr = '';
                                        for($p=0;
$p<sizeof($arrayOfValues); $p++)

                                            $currentPostValue =
$arrayOfValues[$p];
                                            $currentPostValueName
= $currentPostValue->getname();
                                            $currentpostvaluevalue
= $currentpostvalue->getvalue();
```

```
                                                $totalteststr .=
$currentpostvaluename;
                                                $totalteststr .= '=';
                                                $totalteststr .=
$currentpostvaluevalue;

                                                if( $p !=
(sizeof($arrayofvalues) - 1) )
                                                    $totalteststr .=
'&';



                                            $currentformmethod =
strtolower($currentformmethod);

                                            echo 'square injection
present!<br>query: ' . Htmlspecialchars($totalteststr) . '<br>';
                                            echo 'technique: ' .
$currentformmethod . '<br>';
                                            echo 'url: ' .
Htmlspecialchars($actionurl) . '<br>';
                                            echo 'error: ' .
$regularexpression . '';
                                            $tablename = 'take a look
at' . $testid;


                                            $query = "select * from
test_results where test_id = $testid and kind = 'sqli' and
technique = '$currentformmethod' and url = '" .
Addslashes($actionurl) . "' and attack_str = '" .
Addslashes($totalteststr) . "'";
                                            $end result = $db-
>query($query);
                                            if(!$result)
                                                $log->lwrite("could
not execute question $question");
                                            else

                                                $log-
>lwrite("successfully executed question $question");
                                                $numrows = $end
result->num_rows;
                                                if($numrows == 0)

                                                    $log-
>lwrite("number of rows is $numrows for question: $query");

inserttestresult($db, $testid, 'sqli', $currentformmethod,
addslashes($actionurl), addslashes($totalteststr));


                                                $result->loose();
```

```
                                        $http->near();
                                        ruin;



                        $http->close();

                    if(strlen($blunders))
                        echo "<H2 align="center">error:
",$blunders,"</H2>n";


                else if($currentformmethod == 'post')


                    if($urlofsite[strlen($urlOfSite)-1] == '/')
                        $actionurl = $urlofsite .
$currentformaction;
                    else
                        $actionurl = $urlofsite . '/' .
$currentformaction;

                    $blunders=$http-
>getrequestarguments($actionurl,$arguments);

                    $arguments["RequestMethod"]="publish";
                    $arguments["PostValues"]= array();
                    for($p=zero; $p<sizeof($arrayOfValues); $p++)

                        $currentPostValue = $arrayOfValues[$p];
                        $currentPostValueName = $currentPostValue-
>getname();
                        $currentpostvaluevalue =
$currentpostvalue->getvalue();

                        $temparray =
array($currentpostvaluename=>$currentpostvaluevalue);

                        $arguments["PostValues"] =
array_merge($arguments["PostValues"], $temparray);


                    $error=$http->open($arguments);
                    $log->lwrite("url to be asked is:
$actionurl");

                    if($errors=="")

                        $log->lwrite("sending http request to
$actionurl");
                        $mistakes=$https->sendrequest($arguments);

                        if($error=="")
```

```
                            $headers=array();
                            $mistakes=$http-
>readreplyheaders($headers);
                            if($errors=="")

                                $error = $https-
>readwholereplybody($frame);

                                if(strlen($error) == zero)

                                    $vulnerabilityfound = false;
                                    for($warningindex=zero;
$warningindex < sizeof($arrayOfSQLWarnings); $warningIndex++)

                                        $regularExpression =
"/$arrayOfSQLWarnings[$warningIndex]/";


if(preg_match($regularExpression,$body))

                                            $log->lwrite("observed
normal expression: $regularexpression, in body of http response");
                                            $vulnerabilityfound =
genuine;
                                            spoil;



                                    if($vulnerabilityfound)



                                        $totalteststr = '';
                                        for($p=zero;
$p<sizeof($arrayOfValues); $p++)

                                            $currentPostValue =
$arrayOfValues[$p];
                                            $currentPostValueName
= $currentPostValue->getname();
                                            $currentpostvaluevalue
= $currentpostvalue->getvalue();


                                            $totalteststr .=
$currentpostvaluename;
                                            $totalteststr .= '=';
                                            $totalteststr .=
$currentpostvaluevalue;


                                            if( $p !=
(sizeof($arrayofvalues) - 1) )
                                                $totalteststr .=
'&';
```

```
                                        $currentformmethod =
strtolower($currentformmethod);

                                        echo 'square injection
present!<br>query: ' . Htmlspecialchars($totalteststr) . '<br>';
                                        echo 'approach: ' .
$currentformmethod . '<br>';
                                        echo 'url: ' .
Htmlspecialchars($actionurl) . '<br>';
                                        echo 'errors: ' .
$regularexpression . '';

                                        $tablename = 'test' .
$testid;

                                        $query = "choose * from
test_results in which test_id = $testid and kind = 'sqli' and
method = '$currentformmethod' and url = '$actionurl' and
attack_str = '" . Addslashes($totalteststr) . "'";
                                        $end result = $db-
>query($question);
                                        if(!$result)
                                            $log->lwrite("tidak
dapat menjalankan query $question");
                                        else

                                            $log->lwrite("berhasil
dieksekusi question $question");
                                            $numrows = $result-
>num_rows;
                                            if($numrows == 0)

                                                $log-
>lwrite("jumlah baris adalah $numrows dari query: $query");

inserttestresult($db, $testid, 'sqli', $currentformmethod,
$actionurl, addslashes($totalteststr));


                                            $end result-
>unfastened();


                                        $http->near();
                                        spoil;




                        $https->near();

                    if(strlen($error))
                        echo "<H2 align="center">errors:
",$errors,"</H2>n";
```

```
?>

Testforreflectedxss.php

<?Php

Set_time_limit(0);

Function testForReflectedXSS($urlToCheck, $urlOfSite, $testId)

ConnectToDb($db);

UpdateStatus($db, "Pengujian $urlToCheck terhadap Reflected Cross-
Site Scripting...", $testId);

$log = new Logger();

$log->lfile('logs/eventlogs');

$log->lwrite("memulai fungsi uji refcelted xss pada $urltocheck");

$posturl = $urltocheck;

$posturlpath = parse_url($posturl, php_url_path);

//periksa url tidak merespons dengan kode 5xx

$log->lwrite("memeriksa dari mana kode respon diterima
$urltocheck");

$https = new http_class;

$https->timeout=0;

$https->data_timeout=zero;

$http->user_agent="mozilla/5.Zero (windows nt 6.1; wow64)
applewebkit/537.21 (khtml, like gecko) chrome/forty
one.Zero.2228.Zero safari/537.21";

$https->follow_redirect=1;

$https->redirection_limit=five;

$https->settestid($testid);
```

```
$errors=$https->getrequestarguments($urltocheck,$arguments);

$error=$https->open($arguments);

$log->lwrite("url yang akan diminta adalah: $urltocheck");

If($errors=="")

      $log->lwrite("mengirim permintaan http ke $urltocheck");
      $blunders=$https->sendrequest($arguments);

      if($error=="")

            $headers=array();
            $blunders=$https->readreplyheaders($headers);
            if($blunders=="")

                  $responsecode = $https->response_status;
                  $log->lwrite("menerima      kode      tanggapan:
$responsecode");
                  if(intval($responsecode)      >=      500      &&
intval($responsecode) <600)

                        $log->lwrite("kode  respon:  $responsecode
diterima dari: $urltocheck");
                        return;

      $https->close();

If(strlen($blunders))

      echo "<H2 align="center">mistakes: ",$blunders,"</H2>n";
      $log->lwrite("error: $mistakes");
```

```
$html = file_get_html($posturl, $testid);


If(empty($html))



      $log->lwrite("masalah mendapatkan konten dari $urltocheck");

      return;



//kirimkan ini

//jika menambahkan string ke array ini, tambahkan string yang sesuai (untuk dicari sebagai respons), dengan indeks yang sama, dalam array di bawah ini

//respon yang dicari bisa sama dengan payload atau berbeda.

$payloads = array ('<script>alert("hack by zaid mustofa");</script>');


//tanggapan setelah mengirimkan payload yang sesuai dengan ini

$harmfulresponses = array ('<script>alert("hack via zaid mustofa");</script>');


//pertama, periksa apakah url yang diteruskan ke fungsi ini berisi parameter dan mengirimkan payload sebagai parameter tersebut jika ya

$parsedurl = parse_url($urltocheck);

$log->lwrite("periksa apakah $urltocheck berisi parameter");

If($parsedurl)


      if(isset($parsedurl['query']))


            $log->lwrite("$urltocheck memang mengandung parameter");


            $scheme = $parsedurl['scheme'];
```

```
$host = $parsedurl['host'];
$route = $parsedurl['path'];

$query = $parsedurl['query'];
parse_str($query,$parameters);
$originalquery = $question;

$payloadindex = 0;

foreach($payloads as $currentpayload)

      $https = new http_class;
      $https->timeout=0;
      $https->data_timeout=zero;

      $https->user_agent="mozilla/5.0 (windows nt 6.1;
wow64)  applewebkit/537.21  (khtml,  like  gecko)  chrome/forty
one.Zero.2228.Zero safari/537.21";
      $https->follow_redirect=1;
      $https->redirection_limit=5;
      $https->settestid($testid);

      foreach($parameters as $para)

            $query = $originalquery;

            $newquery      =      str_replace($para,
$currentpayload, $query);
            $query = $newquery;

            $testurl  =  $scheme  .  '://'  .  $host  .
$direction . '?' . $question;

            $log->lwrite("url yang akan diminta adalah:
$testurl");
```

```
                        $errors=$https-
>getrequestarguments($testurl,$arguments);


                        $blunders=$http->open($arguments);

                        echo "<br>mengirim permintaan http ke " .
Htmlspecialchars($testurl) . "<br>";

                        if($mistakes=="")


                              $log->lwrite("mengirim    permintaan
http ke $testurl");


                              $errors=$http-
>sendrequest($arguments);


                              if($mistakes=="")


                                    $headers=array();

                                    $blunders=$https-
>readreplyheaders($headers);

                                    if($error=="")


                                          $blunders    =    $https-
>readwholereplybody($body);


                                          if(strlen($blunders)  ==
zero)


                                                $indicatorstr    =
$harmfulresponses[$payloadIndex];

                                                if(stripos($frame,
$indicatorstr))



                                                      echo
'<br>contemplated       xss       gift!<br>query:       '       .
Htmlspecialchars($urltocheck) . '<br>';
```

```
                                                    echo
'approach: get <br>';
                                                    echo 'url: '
. Htmlspecialchars($testurl) . '<br>';
                                                    echo
'blunders: ' . Htmlspecialchars($indicatorstr) . '<br>';
                                                    $tablename =
'take a look at' . $testid;



                                                    $sq. = "pick
out * from test_results where test_id = $testid and type = 'rxss'
and approach = 'get' and url = '$testurl' and attack_str = '" .
Addslashes($query) . "'";
                                                    $end  result
= $db->question($square);
                                                    if(!$result)
                                                          $log-
>lwrite("tidak dapat menjalankan query $sq.");
                                                    else


                                                          $log-
>lwrite("berhasil menjalankan query $sq.");

      $numrows = $end result->num_rows;

      if($numrows == 0)



      $log->lwrite("jumlah  baris  adalah  $numrows  dari  question:
$sq.");

      inserttestresult($db,  $testid,  'rxss',  'get',  $testurl,
addslashes($question));



                                                    $http-
>close();
                                                    break 2;
```

```
                        $http->close();


                    if(strlen($errors))
                        echo    "<H2    align="center">error:
",$errors,"</H2>n";



                $payloadindex++;


Else
    $log->lwrite("tidak   dapat   mengurai   url   yang   rusak:
$urltocheck");



$arrayofforms = array();


$arrayofinputfields = array();


$log->lwrite("mencari $posturl dari forms");


$formnum = 1;
Foreach($html->find('form') as $shape)


    isset($shape->attr['id']) ? $formid = htmlspecialchars($form-
>attr['id']) : $formid = '';
    isset($shape->attr['name'])       ?       $formname       =
htmlspecialchars($form->attr['name']) : $formname ='';
    isset($shape->attr['method'])      ?      $formmethod      =
htmlspecialchars($form->attr['method']) : $formmethod = 'get';
    isset($form->attr['action'])      ?      $formaction      =
htmlspecialchars($shape->attr['action']) : $formaction = '';


    $formmethod = strtolower($formmethod);
```

```
      if(empty($formaction))


            $strlengthurl = strlen($urltocheck);

            $strlengthsite = strlen($urlofsite);

            $firstindexofslash    =    strpos($urltocheck,    '/',
$strlengthsite-1);

            $formaction           =           substr($urltocheck,
$firstindexofslash+1, $strlengthurl);



      $log->lwrite("ditemukan shape di $posturl: $formid $formname
$formmethod $formaction $formnum");


      $newform   =   new   shape($formid,   $formname,   $formmethod,
$formaction, $formnum);

      array_push($arrayofforms, $newform);


      foreach($form->discover('enter') as $enter)


            isset($enter->attr['id'])      ?      $inputid      =
htmlspecialchars($input->attr['id']) : $inputid = '';

            isset($input->attr['name'])     ?     $inputname     =
htmlspecialchars($input->attr['name']) : $inputname = '';

            isset($input->attr['value'])    ?    $inputvalue    =
htmlspecialchars($enter->attr['value']) : $inputvalue = '';

            isset($enter->attr['type'])     ?     $inputtype     =
htmlspecialchars($input->attr['type']) : $inputtype = '';


            $log->lwrite("menemukan   enter   field   di   $posturl:
$inputid $inputname $formid $formname $inputvalue $inputtype
$formnum");


            $inputfield = new inputfield($inputid, $inputname,
$formid, $formname, $inputvalue, $inputtype, $formnum);


            array_push($arrayofinputfields, $inputfield);
```

```
      $formnum ++;



//at this degree, we have to have captured all forms and their
inputs into the corresponding arrays
$log->lwrite('memulai pengujian untuk paperwork');


For($i=0; $i<sizeof($arrayOfForms); $i++)


      $currentForm = $arrayOfForms[$i];
      $currentFormId = $currentForm->getid();
      $currentformname = $currentform->getname();
      $currentformmethod = $currentform->getmethod();
      $currentformaction = $currentform->getaction();
      $currentformnum = $currentform->getformnum();


      $arrayofcurrentformsinputs = array();


      $log->lwrite("memulai   pengujian   shape   pada   $posturl:
$currentformid        $currentformname        $currentformmethod
$currentformaction");


      for($j=zero; $j<sizeof($arrayOfInputFields); $j++)


            $currentInput = $arrayOfInputFields[$j];
            $currentInputIdOfForm = $currentInput->getidofform();
            $currentinputnameofform       =       $currentinput-
>getnameofform();
            $currentinputformnum = $currentinput->getformnum();


            if($currentformnum == $currentinputformnum)


                  array_push($arrayofcurrentformsinputs,
$currentinput);
```

```
      $log->lwrite("memulai pengujian enter fields untuk shape di
$posturl:   $currentformid   $currentformname   $currentformmethod
$currentformaction");

      for($okay=0;       $okay<sizeof($arrayofcurrentformsinputs);
$okay++)


            for($plindex   =   0;   $plindex   <   sizeof($payloads);
$plIndex++)


                  $testStr = $payloads[$plIndex];

                  $log->lwrite("mengirimkan payload: $teststr");

                  $defaultstr = 'diaz123';

                  $indicatorstr = $harmfulresponses[$plIndex];


                  $currentforminput                                 =
$arrayofcurrentformsinputs[$k];

                  $currentforminputname   =     $currentforminput-
>getname();

                  $currentforminputtype   =     $currentforminput-
>gettype();

                  $currentforminputvalue   =    $currentforminput-
>getvalue();


                  if($currentforminputtype!= 'reset')


                        $https = new http_class;

                        $https->timeout=0;

                        $https->data_timeout=0;


                        $https->user_agent="mozilla/5.Zero
(windows nt 6.1; wow64) applewebkit/537.21 (khtml, like gecko)
chrome/forty one.0.2228.0 safari/537.21";

                        $https->follow_redirect=1;

                        $https->redirection_limit=five;

                        $https->settestid($testid);
```

```
                        $arrayofvalues = array();


                        $otherinputs = array();


                        for($l=zero;
$l<sizeof($arrayOfCurrentFormsInputs); $l++)


                                if($currentFormInput->getname()   !=
$arrayofcurrentformsinputs[$l]->getname())


                                        array_push($otherinputs,
$arrayofcurrentformsinputs[$l]);




                        $postobject             =             new
postorgetobject($currentforminputname, $teststr);


                        array_push($arrayofvalues, $postobject);


                        for($m=0; $m<sizeof($otherInputs); $m++)


                                $currentOther = $otherInputs[$m];
                                $currentOtherType  =  $currentOther-
>gettype();
                                $currentothername  =  $currentother-
>getname();
                                $currentothervalue  =  $currentother-
>getvalue();


                                if($currentothertype    ==    'text
$currentothertype == 'password')


                                        $postobject       =       new
postorgetobject($currentothername, $defaultstr);
```

```
                                    array_push($arrayofvalues,
$postobject);
                                $currentothertype == 'post')

                                    $postobject          =     new
postorgetobject($currentothername, $currentothervalue);
                                    array_push($arrayofvalues,
$postobject);

                                else     if($currentothertype     ==
'radio')

                                    $postobject          =     new
postorgetobject($currentothername, $currentothervalue);

                                    $determined = fake;
                                    for($n     =     0;     $n     <
sizeof($arrayOfValues); $n++)

                                        if($arrayOfValues[$n]-
>getname() == $postobject->getname())

                                            $observed        =
proper;
                                            spoil;

                                    if(!$found)
      array_push($arrayofvalues, $postobject);


                        echo '<br><br>';
                        if($currentformmethod == 'get')
                              if($urlofsite[strlen($urlOfSite)-1]
== '/')
                                    $actionurl   =   $urlofsite   .
$currentformaction;
                              else
```

```
                                    $actionurl = $urlofsite . '/'
. $currentformaction;
                              $totalteststr = '';
                              for($p=0; $p<sizeof($arrayOfValues);
$p++)

                                    $currentPostValue            =
$arrayOfValues[$p];
                                    $currentPostValueName        =
$currentPostValue->getname();
                                    $currentpostvaluevalue       =
$currentpostvalue->getvalue();

                                    $totalteststr                .=
$currentpostvaluename;
                                    $totalteststr .= '=';
                                    $totalteststr                .=
$currentpostvaluevalue;

                                    if(           $p             !=
(sizeof($arrayofvalues) - 1) )
                                          $totalteststr .= '&';

                              if(strpos($actionurl, '?')!==false)
                                    $actionurl .= '&';
                              else
                                    $actionurl .= '?';

                              $actionurl .= $totalteststr;

                              $error=$http-
>getrequestarguments($actionurl,$arguments);

                              $blunders=$http->open($arguments);
                              if($error=="")
```

```
                                        $errors=$http-
>sendrequest($arguments);

                                        if($blunders=="")

                                              $headers=array();
                                              $mistakes=$http-
>readreplyheaders($headers);
                                              if($error=="")

                                                    $error   =   $http-
>readwholereplybody($frame);
                                                    if(strlen($error)
== 0)

      if(stripos($body, $indicatorstr))


      $totalteststr = '';

      for($p=0; $p<sizeof($arrayOfValues); $p++)


      $currentPostValue = $arrayOfValues[$p];

      $currentPostValueName = $currentPostValue->getname();

      $currentpostvaluevalue = $currentpostvalue->getvalue();


      $totalteststr .= $currentpostvaluename;

      $totalteststr .= '=';

      $totalteststr .= $currentpostvaluevalue;
```

```
      if( $p != (sizeof($arrayofvalues) - 1) )

      $totalteststr .= '&';



                                                              echo
'pondered       xss       present!<br>question:       '       .
Htmlspecialchars($totalteststr) . '<br>';
                                                              echo
'technique: ' . $currentformmethod . '<br>';
                                                              echo
'url: ' . Htmlspecialchars($actionurl) . '';

      $tablename = 'take a look at' . $testid;



      $question = "pick * from test_results in which test_id =
$testid and kind = 'rxss' and technique = '$currentformmethod' and
url = '$actionurl' and attack_str = '$totalteststr'";
                                                              $end
result = $db->question($query);

      if(!$end result)

      $log->lwrite("could not execute query $query");
                                                              else



      $log->lwrite("efficaciously executed question $query");

      $numrows = $result->num_rows;

      if($numrows == 0)



      $log->lwrite("range   of   rows   is   $numrows   for   question:
$query");
```

```
      inserttestresult($db, $testid, 'rxss', $currentformmethod,
$actionurl, $totalteststr);


                                                                  $http-
>near();
                                                                  smash;


                                        $http->near();


                                  if(strlen($error))
                                        echo                      "<H2
align="center">blunders: ",$errors,"</H2>n";


                            else if($currentformmethod == 'publish')



                                  if($urlofsite[strlen($urlOfSite)-1]
== '/')
                                        $actionurl   =   $urlofsite   .
$currentformaction;
                                  else
                                        $actionurl = $urlofsite . '/'
. $currentformaction;


                                  $blunders=$http-
>getrequestarguments($actionurl,$arguments);



      $arguments["RequestMethod"]="submit";
                                  $arguments["PostValues"]= array();
                                  for($p=zero;
$p<sizeof($arrayOfValues); $p++)


                                        $currentPostValue           =
$arrayOfValues[$p];
                                        $currentPostValueName       =
$currentPostValue->getname();
```

```
                                        $currentpostvaluevalue       =
$currentpostvalue->getvalue();

                                        $temparray                   =
array($currentpostvaluename=>$currentpostvaluevalue);

                                        $arguments["PostValues"]     =
array_merge($arguments["PostValues"], $temparray);



                                $mistakes=$http->open($arguments);

                                if($errors=="")

                                        $blunders=$http-
>sendrequest($arguments);

                                        if($mistakes=="")

                                                $headers=array();
                                                $errors=$http-
>readreplyheaders($headers);
                                                if($mistakes=="")

                                                        $error   =   $http-
>readwholereplybody($frame);
      if(strlen($blunders) == zero)


      if(stripos($frame, $indicatorstr))

      $totalteststr = '';

      for($p=0; $p<sizeof($arrayOfValues); $p++)


      $currentPostValue = $arrayOfValues[$p];
```

```
      $currentPostValueName = $currentPostValue->getname();

      $currentpostvaluevalue = $currentpostvalue->getvalue();


      $totalteststr .= $currentpostvaluename;

      $totalteststr .= '=';

      $totalteststr .= $currentpostvaluevalue;


      if( $p != (sizeof($arrayofvalues) - 1) )

      $totalteststr .= '&';


            echo    'reflected    xss    gift!<br>query:    '    .
Htmlspecialchars($totalteststr) . '<br>';

                                                                  echo
'technique: ' . $currentformmethod . '<br>';

                                                                  echo
'url: ' . Htmlspecialchars($actionurl) . '';

      $tablename = 'test' . $testid;


                                                                  $query
= "choose * from test_results where test_id = $testid and type =
'rxss' and method = '$currentformmethod' and url = '$actionurl' and
attack_str = '$totalteststr'";

      $result = $db->question($question);

      if(!$end result)

      $log->lwrite("tidak bisa mengeksekusi query $query");

                                                                  else


      $log->lwrite("berhasil mengeksekusi query $question");
```

```
      $numrows = $end result->num_rows;

      if($numrows == zero)



      $log->lwrite("jumlah baris adalah $numrows untuk question:
$question");

      inserttestresult($db, $testid, 'rxss', $currentformmethod,
$actionurl, $totalteststr);



                                                                  $http-
>near();

      damage;


                                      $http->near();

                                if(strlen($errors))

                                      echo                        "<H2
align="center">mistakes: ",$errors,"</H2>n";

?>
```

Databsefunctions.php

```
<?Php

Function connectToDb(&$db)

    $db = $db = new mysqli( 'localhost', 'root', '', 'webscan');
    if (mysqli_connect_errno())
        return false;

    return true;


//Update status of test in db
//e.G. UpdateStatus($db, 'Starting scan...', 1234);
//Returns true on success, False on failure.
Function updateStatus($db, $newStatus, $testId)

    $query = "UPDATE tests SET status = '$newStatus' WHERE id =
$testId;";
    $result = $db->question($query);
    go back $result;
```

```
Feature inserttestresult($db, $testid, $type, $method, $url,
$attackstr)

    $question = "insert into test_results(test_id, kind, approach,
url, attack_str)
values($testid,'$type','$technique','$url','$attackstr')";
    $end result = $db->question($question);
    go back $end result;


//generates the following check identity
//return the next check identification on fulfillment. Otherwise
returns false.
Characteristic generatenexttestid($db)

    $question = "pick out max(identification) from exams";
    $result = $db->question($query);
    if(!$result)
        go back $result;

    $row = $end result->fetch_array();

    $maxid = $row[0] + 1;
    //$maxid = $row->identity;//in any other case $row-
>max(identification)
    return $maxid;


//adds 1 to the present day variety of http requests sent
//returns proper on fulfillment, false on failure
Characteristic incrementhttprequests($db, $testid)

    $question = "update checks set num_requests_sent =
(num_requests_sent + 1) in which id = $testid";
    $result = $db->query($question);
    return $end result;


?>


Createpdfreport.php


<?Php

Function createPdfReport($testId, &$fileName)

ConnectToDb($db);
UpdateStatus($db, "Generating PDF report for test: $testId...",
$testId);

$log = new Logger();
$log->lfile('logs/eventlogs');
```

```
$log->lwrite("beginning pdf generator function for test:
$testid");

// create new pdf file
$pdf = new tcpdf(pdf_page_orientation, pdf_unit, pdf_page_format,
authentic, 'utf-8', false);

// set record records
$pdf->setcreator(pdf_creator);
$pdf->setauthor('webscanner');
$pdf->settitle('document for test: ' . $testid);
$pdf->setsubject('terdeteksi kerentanan');

// set default header facts
Date_default_timezone_set('wib');
//$now = date('l js f y h:i:s a');
//$headerstr = "test id: $testidn$now";
//$pdf->setheaderdata(pdf_header_logo, pdf_header_logo_width,
'internet site vulnerability scaner', $headerstr);

// set header and footer fonts
$pdf->setheaderfont(array(pdf_font_name_main, '',
pdf_font_size_main));
$pdf->setfooterfont(array(pdf_font_name_data, '',
pdf_font_size_data));

// set default monospaced font
$pdf->setdefaultmonospacedfont(pdf_font_monospaced);

//set margins
$pdf->setmargins(pdf_margin_left, pdf_margin_top,
pdf_margin_right);
$pdf->setheadermargin(pdf_margin_header);
$pdf->setfootermargin(pdf_margin_footer);

//set vehicle web page breaks
$pdf->setautopagebreak(true, pdf_margin_bottom);

//set image scale aspect
$pdf->setimagescale(pdf_image_scale_ratio);

//set some language-established strings
International $l;
$pdf->setlanguagearray($l);

// ---------------------------------------------------------

// set default font subsetting mode
$pdf->setfontsubsetting(authentic);

// set font
// dejavusans is a utf-eight unicode font, in case you best want
to
// print popular ascii chars, you can use core fonts like
// helvetica or instances to reduce report length.
$pdf->setfont('dejavusans', '', 10, '', proper);
```

```
// add a page
// this technique has several alternatives, take a look at the
source code documentation for extra statistics.
$pdf->addpage();


//generate summary
$log->lwrite("displaying precis in pdf");
$summary = '';
$question = "pick * from checks where identification = $testid";
$result = $db->question($question);
If(!$end result)
    $log->lwrite("could not execute question $query");
Else

    $log->lwrite("efficaciously completed question $query");
    $row = $end result->fetch_object();

    $urlsfound = $row->numurlsfound;
    $requestssent = $row->num_requests_sent;
    $starttime = $row->start_timestamp;
    $fintime = $row->finish_timestamp;
    $targetsite = $row->url;

    $starttimeformatted = date('l js f y h:i:s a', $starttime);
    $fintimeformatted = date('l js f y h:i:s a', $fintime);
    $period = $fintime - $starttime;
    $minutes = intval($length/60);
    $seconds = $duration % 60;
    $secondsstr = strval($seconds);
    $secondsformatted = str_pad($secondsstr,2,"0",str_pad_left);

    $question = "pick out * from test_results where test_id =
$testid;";
    $end result = $db->query($query);
    $numvulns = 0;
    if($result)
        $numvulns = $result->num_rows;
    else
        $log->lwrite("couldn't execute query $query");

    //populate vulnerability kinds right into a list to be used
when calculating pie chart dimensions
    $vulntypes = array();
    for($i=zero; $i<$numVulns; $i++)

        $row = $result->fetch_object();
        $kind = $row->kind;
        array_push($vulntypes, $type);


    $summary .= '<table>';
    $summary .= "<tr><td>target
webiste:</td><td>$targetsite</td></tr>";
```

```
    $precis .= "<tr><td>lama scan :</td><td>$secondsformatted
seconds</td></tr>";

    $summary .= "<tr><td>jumlah
url:</td><td>$urlsfound</td></tr>";
    $precis .= "<tr><td>jumlah
kerentanan:</td><td>$numvulns</td></tr>";
    $precis .= "<tr><td>jumlah request
http:</td><td>$requestssent</td></tr>";
    $precis .= '</table>';



$html = '<h2>summary</h2>' . $precis;

$pdf->writehtmlcell($w=0, $h=zero, $x='', $y='', $html,
$border=zero, $ln=1, $fill=zero, $reseth=actual, $align='',
$autopadding=genuine);



If($numvulns > 0)

    //generate details of vulnerabilities found
    $html = '<h3>terdeteksi kerentanan</h3><br>';
    $pdf->writehtmlcell($w=0, $h=0, $x='', $y='', $html,
$border=0, $ln=1, $fill=0, $reseth=real, $align='',
$autopadding=true);

    //become aware of what vulnerabilities were observed
    $log->lwrite("figuring out what vulnerabilities were
determined at some stage in check");
    $vulnsfound = array();//array containing vulnerability items
of all vulnerabilities found for this check
    $vulnsids = array();//array containing the ids of the flaws
determined (and not using a duplications) for this take a look at
    $query = "choose * from test_results where test_id = $testid";
    $result = $db->query($question);
    if(!$result)
        $log->lwrite("couldn't execute query $query");
    else

        $log->lwrite("successfully carried out query $question");
        $numrows = $end result->num_rows;
        for($i = zero; $i < $numRows; $i++)

            $row = $result->fetch_object();

            $test_id = $row->test_id;
            $kind = $row->kind;
            $method = $row->method;
            $url = $row->url;
            $attack_str = $row->attack_str;

            $vuln = new vulnerability($test_id, $kind, $method,
$url, $attack_str);
```

```
            array_push($vulnsfound, $vuln);

            if(!In_array($kind, $vulnsids))
                array_push($vulnsids, $type);



    usort($vulnsids, "comparevulns");

    //displaying information of every vulnerability found which
includes description,
    //solution, priority and showing all times where it changed
into discovered
    $log->lwrite("displaying details in pdf of every vulnerability
determined");

    foreach($vulnsids as $currentid)

        $html = '';
        $query = "pick out * from vulnerabilities where identity =
'$currentid';";
        $result = $db->query($query);
        if(!$result)
            $log->lwrite("could not execute question $query");
        else

            //show details of vulnerability
            $row = $result->fetch_object();
            $name = $row->name;
            $description = $row->description;
            //$answer = $row->solution;
            //$priority = $row->precedence;
            $html .= "<h3>$call</h3>";
            //$html .= "<h4>precedence: </h4>$priority";
            $html .= "<h4>description: </h4>";
            $html .= stripslashes($description);
            //$html .= "<h4>tips: </h4>";
            $html .= stripslashes($answer);
            $html .= '<br>';
            $html .= '<h4>terletak pada:</h4>';

            //display all times of vulnerability
            foreach($vulnsfound as $currentvuln)

                if($currentvuln->gettype() == $currentid)

                    $html .= '<b>url:</b> ' .
Htmlspecialchars($currentvuln->geturl()) . '<br>';
                    $html .= '<b>method:</b> ' .
Strtoupper($currentvuln->getmethod()) . '<br>';

                    $type = $currentvuln->gettype();
                    $attackstr = htmlspecialchars($currentvuln-
>getattackstr());
```

```
                    if($kindkind == 'sxss' type == 'sqli' type ==
'basqli')
                        $html .= "<b>query used:</b>
$attackstr<br>";
                    $html .= '<br>';



        $html .= '<br><br>';
        //echo $html;
        $pdf->writehtmlcell($w=0, $h=0, $x='', $y='', $html,
$border=0, $ln=1, $fill=0, $reseth=genuine, $align='',
$autopadding=actual);

        $html = '';


Else
    $html = '<h2>tidak ditemukan kerentanan</h2><br>';

$html .= '<h1>hatur nuhun</h1>';
$pdf->writehtmlcell($w=0, $h=zero, $x='', $y='', $html,
$border=zero, $ln=1, $fill=zero, $reseth=genuine, $align='',
$autopadding=genuine);

$filename = 'reviews/test_' . $testid . '.Pdf';

//output pdf, this feature has a couple of alternatives
$pdf->output($filename, 'f'); //set this to 'f' to shop as report,
'i' to output to browser, e: return the report as base64 mime
multi-component email attachment
//$pdf->output('take a look at.Pdf', 'i');//for testing


//compares  vulnerability ids based on their priority
//this function is handed into Hypertext Preprocessor's usort
feature at the side of an array of vulnerability ids
Characteristic comparevulns($vulnoneid, $vulntwoid)

    if(!Connecttodb($db))
        return 0;
    $queryone = "pick * from vulnerabilities wherein
identification = '$vulnoneid'";
    $querytwo = "pick * from vulnerabilities where id =
'$vulntwoid'";
    $resultone = $db->question($queryone);
    $resulttwo = $db->question($querytwo);

    if(!($resultone && $resulttwo))
        return 0;

    $rowone = $resultone->fetch_object();
    $rowtwo = $resulttwo->fetch_object();

    $vulnonepriority = $rowone->priority_num;
    $vulntwopriority = $rowtwo->priority_num;
```

```
    if($vulnonepriority == $vulntwopriority)
      return 0;
    else if($vulnonepriority > $vulntwopriority)
        return 1;;
    else //$vulnonepriority < $vulnTwoPriority
        return 1;


?>
```