https://doi.org/10.37817/tekinfo.v25i1
P-ISSN: 1411-3635
E-ISSN: 2962-5645

Analisis pengamanan *Website* dari Serangan *Cross Site Script* (XSS) dengan *htmlspecialchars* dan *strip_tags*

[1]Fadly Maulana Adiyasa, [2]Nabil Lail Alamsyah, [3]Hizrawan Mohammad Nur Liputo, [4]Essy Malays Sari Sakti
[1,2,3,4]Teknik Informatika, Universitas Persada Y.A.I, Jakarta Pusat

E-mail: [1]fadlyadiyasa9@gmail.com, [2]nabillail78@gmail.com, [3]hizrawanliputoo@gmail.com, [4]emalays67@gmail.com

ABSTRAK

Di era yang serba digital ini, kejahatan dalam jaringan menjadi hal yang sering terjadi. Jika dahulu kejahatan terjadi secara fisik, sekarang kejahatan terjadi secara digital. Pencurian, perampokan, penindasan bisa terjadi dalam jaringan. Salah satu kejahatan yang biasa terjadi adalah kejahatan yang menyerang *website*, aplikasi dan sesuatu yang menyimpan data lainnya. Kejahatan ini bisa dilakukan dengan menggunakan teknik *Cross Site Scripting* (XSS). Kejahatan jenis ini memanfaatkan skrip berbahaya yang di masukan ke dalam *database* melalui form yang tersedia pada *website* yang sudah ditargetkan. Kejahatan ini bisa dilakukan untuk tujuan pencurian data atau hanya sekedar kejahilan belaka. Tujuan dari penelitian ini adalah untuk mengamankan *database* dengan program yang ada dan untuk mencegah skrip berbahaya ini berhasil masuk kedalam *database*. Penelitan ini menggunakan metode eksperimen semu yang merupakan metode percobaan guna untuk mendapatkan data yang dibutuhkan dengan terlebih dahulu membuat suatu website sebagai sasarannya uji coba. Percobaan dilakukan dengan mendapat serangan Cross Sote Script (XSS) pada website tanpa pengamanan dan dengan pengamanan. Hasil penelitian memperlihatkan bahwa menambahkan fungsi htmlspecialchars() dan Strip_tags() dapat melindungi website dari skrip berbahaya yang akan dimasukan melalui form input yang tersedia di website.

Serangan Cross Site Script (XSS) dengan htmlspecialchars dan strip_tags

Kata kunci : Digital, Kejahatan, Skrip, Cross Site Scripting, Database, Keamanan, Program

ABSTRACT

In this digital era, online crime has become a frequent occurrence. If previously crime occurred physically, now crime occurs digitally. Theft, robbery, bullying can occur in the network. One of the crimes that commonly occurs is crimes that attack websites, applications and things that store other data. This crime can be committed using Cross Site Scripting (XSS) techniques. This type of crime utilizes malicious scripts that are entered into the database via a form available on the targeted website. This crime can be committed for the purpose of data theft or just for pure mischief. The aim of this research is to secure the database with existing programs and to prevent this malicious script from successfully entering the database. This research uses a quasi-experimental method which is an experimental method to obtain the required data by first creating a website as the target for testing. The experiment was carried out by getting a Cross Sote Script (XSS) attack on websites without security and with security. The research results show that adding the htmlspecialchars() and Strip_tags() functions can protect the website from malicious scripts that will be entered via the input form available on the website.

P-ISSN: 1411-3635
E-ISSN: 2962-5645
https://doi.org/10.37817/tekinfo.v25i1

Keyword : Digital, Crime, Scripts, Cross Site Scripting, Databases, Security, Programs

1. PENDAHULUAN

Latar Belakang

Diera digitalisasi, internet menjadi sangat penting bagi kehidupan sehari-hari. Bagian yang terdapat dalam internet adalah *website*. Suatu *website* dapat digunakan untuk keperluan pribadi, membuat portofolio, perusahaan, penjualan secara online, pendidikan, instansi, komunitas, sosial media dan hiburan. *Website* juga bisa diperbarui secara berkala dan ada juga yang tidak. Karna itu, *website* tidak luput dari kejahatan. Kejahatan yang tadinya berbentuk fisik kini berubah menjadi digital. Para *hacker* bisa menyerang suatu *website* dengan berbagai cara untuk mencuri data, mengacak-acak website demi mendapatkan keuntungan baik berupa uang ataupun kesenangan pribadi. Salah satu serangan *cyber* yang menyerang *webiste* adalah *Corss Site Script*(XSS). XSS bekerja dengan menambahkan skrip tertentu untuk kemudian disimpan dalam *database* korban. Skrip yang sudah disimpan akan digunakan terus menerus selama ada yang membuka web tersebut. Skrip ini dapat digunakan untuk mencuri *cookie* atau merusak web sesuai dengan keinginan dari penyerang.(Malays, Sakti, & Basry, 2015; Rahmawati, 2017; Widya et al., 2018)

2. LANDASAN TEORI

A. Pengertian *Website*

Website adalah suatu web yang berjalan dalam internet dan dikenal dengan nama WWW. *World Wide Web* (WWW*)* adalah suatu sistem dalam internet yang digunakan untuk menjalankan suatu format khusus yang berisi tulisan, gambar atau suara. Format ini biasa dijalankan dalam bentuk *HyperText Markup Language* (HTML). Suatu *website* biasa menampilkan program yang sudah dibuat terlebih dahulu. Di dalam program ini terdapat *database* yang disimpan. *Website* dibuat oleh seorang *programer*. Seorang *programer* bisa membuat sebuah *website* sesuai dengan keinginannya menggunakan bahasa yang tersedia. Banyak *tools* yang tersedia agar dapat menjalankan berbagai bahasa pemrograman. Salah satu contoh dari *tools* yang ada adalah Neetbeans, Visual Studio Code. *Website* bisa menampilkan informasi, Informasi ini disimpan dalam sebuah *database* yang dapat diubah, dihapus oleh administrator. (Rohaya, 2008; Titus Aditya Kinaswara, Nasrul Rofi'ah Hidayati, & Fatim Nugrahanti, 2019)

B. Pengertian Database

Database merupakan kumpulan data yang dikelola untuk kemudahan pengelolaannya berdasarkan peraturan tertentu yang saling terhubung. *Database* memungkinkan pengguna untuk dengan mudah mencari, menyimpan dan membuang informasi.(Andaru Andri, 2018; Sofwan, 2003)

C. Pengertian MySQL

MySQL adalah salah satu sistem *open source* untuk pembuatan *database*. Untuk menjalankan MySQL ini bisa menggunakan *tools* bernama XAMPP. XAMPP ini bisa menjalankan beberapa program yang salah satunya MySQL database. XAMPP juga bisa menjalankan program lain seperti Apache, PHP, Perld dan bisa berjalan lintas platform.(Hengki Tamando Sitohang, 2018; Sofwan, 2003)

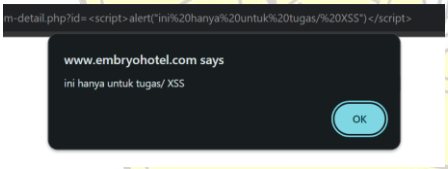
D. Pengertian *Cross Site Script* (XSS)

Cross Site Scripting (XSS) adalah sebuah skrip program yang dibuat oleh *hacker* dengan memanfaatkan halaman web itu sendiri. XSS digunakan oleh penyerang dengan tujuan untuk

https://doi.org/10.37817/tekinfo.v25i1
P-ISSN: 1411-3635
E-ISSN: 2962-5645

mengambil data penting, *cookie* atau membuat suatu program yang dapat merusak *user* itu sendiri, namun dibuat seolah-olah kerusakan terjadi akibat web itu sendiri. XSS bekerja dengan menambahkan skrip tertentu untuk kemudian disimpan dalam *database* korban. Skrip yang sudah disimpan akan digunakan terus menerus selama ada yang membuka web tersebut. Skrip ini dapat digunakan untuk mencuri *cookie* atau merusak web sesuai dengan keinginan dari penyerang. Kerusakan ini bisa dihapus dengan menghapus skrip yang sudah disimpan dalam *database*. Skirp ini dapat dicegah dengan menambahkan pengaman berupa *htmlspecialchars dan strip_tags*. (Hambartsumyan, 2011; Suroto & Asman, 2021)

Kejahatan ini akan terlihat seperti kesalahan pada web itu sendiri. Korban akan sulit menyadari kalau sudah terkena kejahatan *cyber*. Skrip ini akan dijalankan terus menerus sampai administrator menghapus skrip ini dari *database*-nya.



Gambar1.Contoh serangan XSS ringan (Sumber :pribadi)

Kejatahatan *cyber* menggunakan teknik XSS ini dibagi menjadi tiga jenis. Serangan berbentuk *Reflected* XSS, *Stored* XSS dan DOM-*based* XSS.(Indah O. Laleb, 2022)

1. *Reflected* XSS

Reflected XSS adalah serangan XSS dengan memanfaatkan URL yang sudah disisipkan skrip tertentu. Jika dibuka tautannya maka skrip tersebut akan dieksekusi.(Alenzi & Bashir Abbas, 2022; Indah O. Laleb, 2022; Suroto & Asman, 2021)

2. *Stored* XSS

Stored XSS adalah serangan XSS yang terjadi dengan menyisipkan skrip berbahaya kedalam *database website*. Ketika ada seseorang yang mengunjungi *website* tersebut, skrip akan otomatis dijalankan. Penyerang bisa memasukan skrip kedalam *database* melalui form yang tersedia didalam *website* tersebut. Jika *website* memiliki keamanan yang kurang bagus, maka skrip ini akan masuk kedalam *database* dengan mudahnya tanpa disaring terlebih dahulu.(Alenzi & Bashir Abbas, 2022; Indah O. Laleb, 2022; Suroto & Asman, 2021)

3. DOM-*based* XSS

DOM-*based* XSS adalah serangan XSS yang bekerja dengan memanipulasi *Document Object Model* (DOM) dari *website* yang sudah ditergetkan untuk kemudian mengubah strukturnya. DOM-based XSS ini biasa digunakan untuk mencuri *cookie* pengunjung *website*. *Cookie* yang sudah diambil bisa dimanfaatkan oleh pelaku untuk mengambil data-data korban yang tersimpan dalam *website* seperti *username, password*, riwayat.

Skrip ini bisa dicegah dengan menambahkan fungsi *strip_tags* dan *htmlspecialchars* kedalam program yang memiliki fungsi untuk menyimpan *database*. *Strip_tags* akan menghapus seluruh *tag* HTML yang dimasukan kedalam inputan user. *Ht*mlspecialchars akan mengubah karakter khusus menjadi karakter lain yang tidak digunakan dalam penulisan skrip HTML. (Alenzi & Bashir Abbas, 2022; Indah O. Laleb, 2022; Suroto & Asman, 2021)

E. Pengertian *Htmlspecialchars* dan *Strip_tags*

Htmlspecialchars adalah sebuah fungsi yang mengubah suatu karakter spesial dalam HTML menjadi karakter lain. *Stirp_tags* adalah fungsi yang bekerja dengan menghilangkan atau menghapus seluruh *tag* HTML dari semua

P-ISSN: 1411-3635
E-ISSN: 2962-5645
https://doi.org/10.37817/tekinfo.v25i1

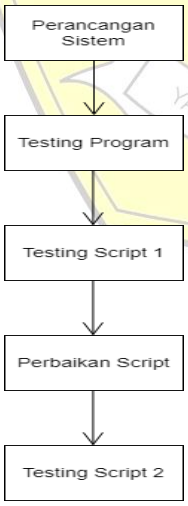
input pengguna. Ini dilakukan untuk menghindari penyalahgunaan karakter dalam memasukan skrip berbahaya kedalam *website*. Karakter yang diubah antara lain:

- & menjadi &
- " menjadi &quote;
- ' menjadi '
- < menjadi <
- menjadi >

dengan begitu skrip berbahaya ini menjadi tidak bisa berjalan sebagaimana semestinya.(Hambartsumyan, 2011; Prayogo & Wahanggara, 2016)

3. METODOLOGI

Metodelogi penelitian ini adalah kualitataif dengan pendekatan metode eksperimen semu dan metode untuk perancangan menggunakan WDLC . WLDC (*Web Development Life Cycle*) adalah suatu metode yang digunakan untuk mengembangkan *website*. Metode eksperimen semu adalah suatu metode yang digunakan untuk menguji suatu data dengan cara melakukan percobaan terlebih dahulu, dalam hal ini adalah pembuatan website.(Kaban, 2017; Sutono & Pamungkas, 2021)



Gambar 2.Langkah Penelitian

A. Perancangan sistem

Pada tahap ini dibuatlah *website* terlebih dahulu. *website* akan dibuat secara sederhana menggunakan Visual Studio Code dan XAMPP. Visual Studio Code akan digunakan untuk membuat tampilan *website* dan fungsi sederhana yang terdiri dari *create, read, update, delete* (CRUD). XAMPP akan digunakan untuk membuat *databasenya*. Tahap ini bertujuan untuk menciptakan *website* yang dapat digunakan untuk memberi contoh serangan beserta pengamanannya.

B. Testing Program

Pada tahap ini akan dilakukan pengujian untuk menjalankan program sederhana pada *website* yang hanya terdiri dari CRUD. *Website* akan diuji fungsi untuk membuat, mengubah dan menghapusnya. Ini bertujuan untuk mengetahui apakah program bisa berjalan sebagaimana semestinya. Jika sudah berhasi pada tahap ini maka bisa menuju tahap selanjutnya.

C. *Testing* Skrip 1

Pada tahap ini dibuatlah skrip berupa <script>alert("tugas kuliah")</script> yang bisa merusak *website* dengan terus menerus memberikan notifikasi berupa *pop up* yang akan mengganggu setiap *website* sedang diakses. Kemudian akan ditambahkan skrip seperti matraman yang bisa mengubah warna inputan sesuai keinginan penulis skrip. Ini bertujuan untuk memberitahukan apa yang terjadi jika skrip berhasil masuk kedalam *database* sistem.

D. Perbaikan

Pada tahap ini akan ditambahkan suatu fungsi yang bisa membatalkan skrip tersebut. Setelah ditambahkan maka skrip yang tidak diingingkan tidak akan masuk kedalam *database* dan membuat *website* menjadi berantakan. Ini bertujuan sebagai bentuk pencegahan jika suatu saat ada orang yang sedang berbuat kejahatan. Skrip yang dimasukan tadi adalah skrip yang bertujuan untuk kejahilan saja. Skrip

https://doi.org/10.37817/tekinfo.v25i1
P-ISSN: 1411-3635
E-ISSN: 2962-5645

tadi bisa saja diubah dan digunakan untuk mencuri *cookie*, data-data pemilik dan pengunjung *website* yang tentunya akan sangat berbahaya jika sampai bocor.

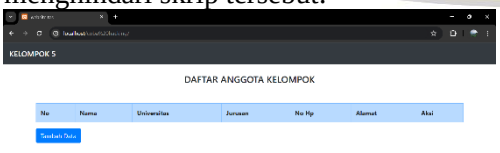
E. *Testing* Skrip 2

Pada tahap ini fungsi khusus yang sudah ditambahkan akan bekerja dengan cara mengubah karakter khusus pada skrip menjadi karakter lain yang tidak berfungsi dalam pembuatan tab pada HTML. Jika melihat data yang disimpan dalam *database*, maka beberapa karakter yang ada dalam skrip akan berubah menjadi karakter lain. Dengan begitu skrip ini jadi tidak bisa berjalan sesuai keinginan penyerang dan hanya akan menjadi teks biasa.

4. HASIL DAN PEMBAHASAN

A. Perancangan Sistem

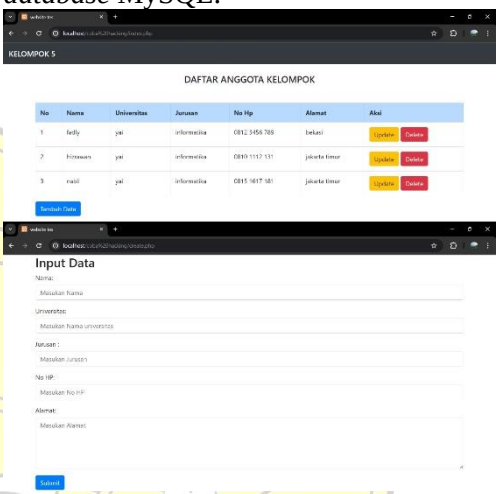
Pada tahapan ini dibuatlah *website* terlebih dahulu. *Website* ini dibuat dengan menggunakan fungsi sederhana berupa *create*, *read*, *update*, *delete* (CRUD). Program dibuat dengan menggunakan Visual Studio Code dan XAMPP untuk pembuatan HTML dan MySQL-nya. Program ini terdiri dari empat *class* yang tiap *calssnya* berfungsi untuk menampilkan antarmuka, koneksi, *update* dan *delete*. *Website* ini akan digunakan sebagai media untuk menjalankan skrip berbahaya dan juga digunakan untuk meberikan contoh pengaman untuk menghindari skrip tersebut.



Gambar 3.*Website*

B. Tes Program

Pada tahap ini *website* diuji terlebih dahulu untuk menjalankan program seperti semestinya. Program sederhana ini bekerja dengan menambahkan data, membaca data, menghapus data, memperbarui data. Data yang sudah ditambahkan akan masuk kedalam *database* MySQL.

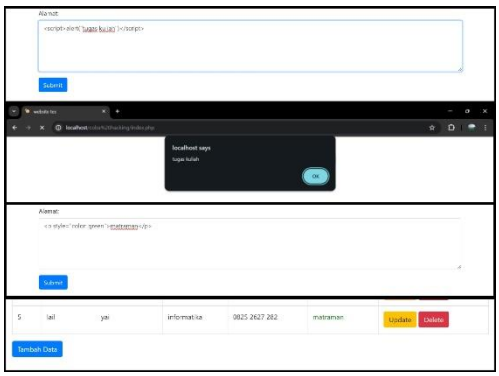


Gambar 4.Tes Program

C. Tes Skrip 1

Pada tahap ini skrip akan ditambahkan. Skrip akan dimasukan kedalam *database* dengan memanfaatkan form yang tersedia pada *website*. Skirp yang dituliskan akan berupa <script>alert(“tugas kuliah”)</script>. Jika skrip ini berhasi masuk kedalam *database*, maka skrip ini akan terus muncul sebagai *pop up* setiap kali ada yang mengakses *website* ini. Kemudian akan ditambahkan skrip bertuliskan matraman .
skrip ini akan mengubah warna dari kata matraman sesuai apa yang dituliskan pada skrip.

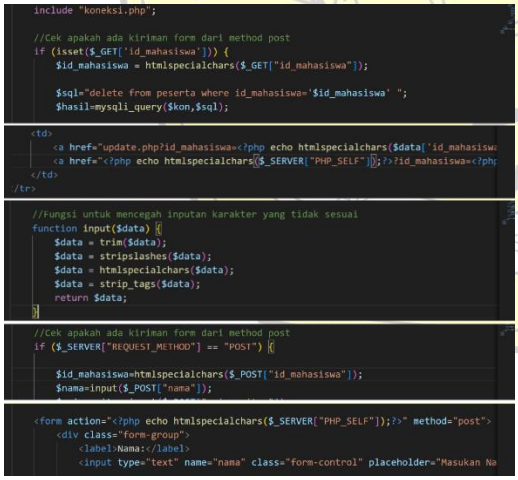
P-ISSN: 1411-3635
E-ISSN: 2962-5645
https://doi.org/10.37817/tekinfo.v25i1



Gambar 5.Tes Skrip 1

D. Perbaikan

Pada tahap ini akan ditambahkan fungsi tertentuk kedalam program untuk menyaring skrip yang tidak diinginkan. Fungsi yang ditambahkan berupa htmlspecialchars dan *strip_tags*. *Htmlspecialchars* dan *strip_tags* akan ditambahkan pada setiap perintah yang berisi input yang memiliki akses kedalam *database*. Dengan begitu, saat ada skrip yang dimasukan melalui form, maka tag yang ada pada skrip tersebut akan diubah fungsinya agar aman dan menjadi tidak berbahaya lagi.

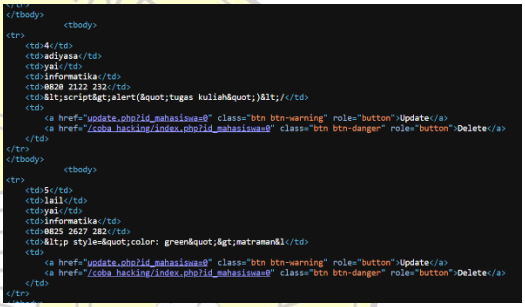


Gambar 6.Perbaikan

E. Tes Skrip 2

Pada tahap ini skrip pertama sudah dihapus dari *database* dan *website* dijalankan seperti seharusnya. Skrip kemudian akan ditulis Kembali kedalam form yang tersedia pada *website*. Form kemudian akan mengirimkan Kembali data kedalam *database*. Program yang sudah diberikan pengaman ini akan mengubah karakter khusus pada skrip ini menjadi karakter lain saat dikirimkan kedalam *database*. Akan tetapi, skrip tetap akan tampil dilayar sebagai semestinya tanpa mengubah karakter khususnya.

Gambar 7.Tes Skrip 2 *Browser*



Gambar 8. Tes Skrip 2 *Page Source*

5. KESIMPULAN

Kesimpulannya adalah salah satu kejahatan *website* berupa pemberian skrip melalui *website* atau *Cross Site Script* (XSS) dapat dicegah degan memberikan pengaman sederhana. Hanya dengan menambahkan fungsi htmlspecialchars() dan Strip_tags() bisa melindungi website dari skrip berbahaya yang akan dimasukan melalui form input yang tersedia di website. Dengan begitu skrip berbahaya tersebut tidak dapat bekerja sebagaimana yang diinginkan penyerang.

6. UCAPAN TERIMA KASIH

Terima kasih kami ucapkan kepada LPPM UPI Y.A.I yang telah memberikan kami kesempatan dan inspirasi dalam pembuatan jurnal ini. Kami juga turut mengucapkan terima kasih kepada rekan-rekan yang sudah memberikan inspirasi

kepada kami dalam proses pembuatan jurnal ini.

DAFTAR PUSTAKA

Alenzi, K. F., & Bashir Abbas, O. A. (2022). A Defensive Framework for Reflected XSS in Client-Side Applications. *Journal of Web Engineering*, *21*(7), 2209–2230. https://doi.org/10.13052/jwe1540-9589.2179

Andaru Andri. (2018). *PENGERTIAN DATABASE SECARA UMUM*.

Hambartsumyan, H. (2011). *Precise XSS Detection with Static Analysis using String Analysis*.

Hengki Tamando Sitohang. (2018). *SISTEM INFORMASI PENGAGENDAAN SURAT BERBASIS WEB PADA PENGADILAN TINGGI MEDAN*.

Indah O. Laleb. (2022). *ANALISIS CROSS-SITE SCRIPTING (XSS) INJECTION-REFLECTED XSS AND STORED XSS MENGGGUNAKAN FRAMEWORK OWASP 10 Indah O. Laleb*. Retrieved from http://attack.com/page.php?something=someth

Kaban, R. (2017). PENGEMBANGAN SISTEM INFORMASI PERPUSTAKAAN DENGAN FRAMEWORK CSS BOOTSTRAP DAN WEB DEVELOPMENT LIFE CYCLE. In *Jurnal Ilmiah Informatika* (Vol. 2). Retrieved from http://getbootstrap.com

Malays, E., Sakti, S., & Basry, A. (2015). *APLIKASI REKENING BERSAMA SEBAGAI MEDIASI PEMBELI-PENJUAL DALAM TRANSAKSI ONLINE STORE*.

Prayogo, R. R., & Wahanggara, V. (2016). *Analisis keamanan menggunakan web aplikasi bwapp terhadap serangan XSS (Cross Site Scripting) dan SQL Injection*.

Rahmawati, I. (2017). *ANALISIS MANAJEMEN RISIKO ANCAMAN KEJAHATAN SIBER (CYBER CRIME) DALAM PENINGKATAN CYBER DEFENSE*. Retrieved from https://news.detik.com/

Rohaya, S. (2008). *INTERNET: PENGERTIAN, SEJARAH, FASILITAS DAN KONEKSINYA*. Retrieved from http://dhani.shingcat.com,

Sofwan, A. (2003). *Belajar Mysql dengan Phpmyadmin*. Retrieved from http://blog.sofwan.net

Suroto, S., & Asman, A. (2021). *ANCAMAN TERHADAP KEAMANAN INFORMASI OLEH SERANGAN CROSS-SITE SCRIPTING (XSS) DAN METODE PENCEGAHANNYA* (Vol. 11). Retrieved from http://www.hackers.com?yid=

Sutono, S., & Pamungkas, A. P. (2021). Penerapan Metode Eksperimen Semu Pada Sistem Informasi Persediaan dan Penjualan Obat di Apotek Berbasis Web-Base. *Media Jurnal Informatika*, *12*(2), 44. https://doi.org/10.35194/mji.v12i2.1225

Titus Aditya Kinaswara, Nasrul Rofi'ah Hidayati, & Fatim Nugrahanti. (2019). *Rancang Bangun Aplikasi Inventaris Berbasis Website pada Kelurahan Bantengan*.

Widya, N., Dosen, S. ", Pamulang, U., Surya Kencana, J., Pamulang, S., & Selatan, T. (2018). KEJAHATAN CYBER DALAM PERKEMBANGAN TEKNOLOGI INFORMASI BERBASIS KOMPUTER. In *Jurnal Surya Kencana Dua: Dinamika Masalah Hukum dan Keadilan* (Vol. 5).