

Modern Frontend Fejlesztés

2021 - One Identity

Rólunk

- One Identity
- Cybersecurity
- PAM piac – Privileged Access Management
- Magyar részleg főként Networking érdeklődéssel, logolással



Rólunk

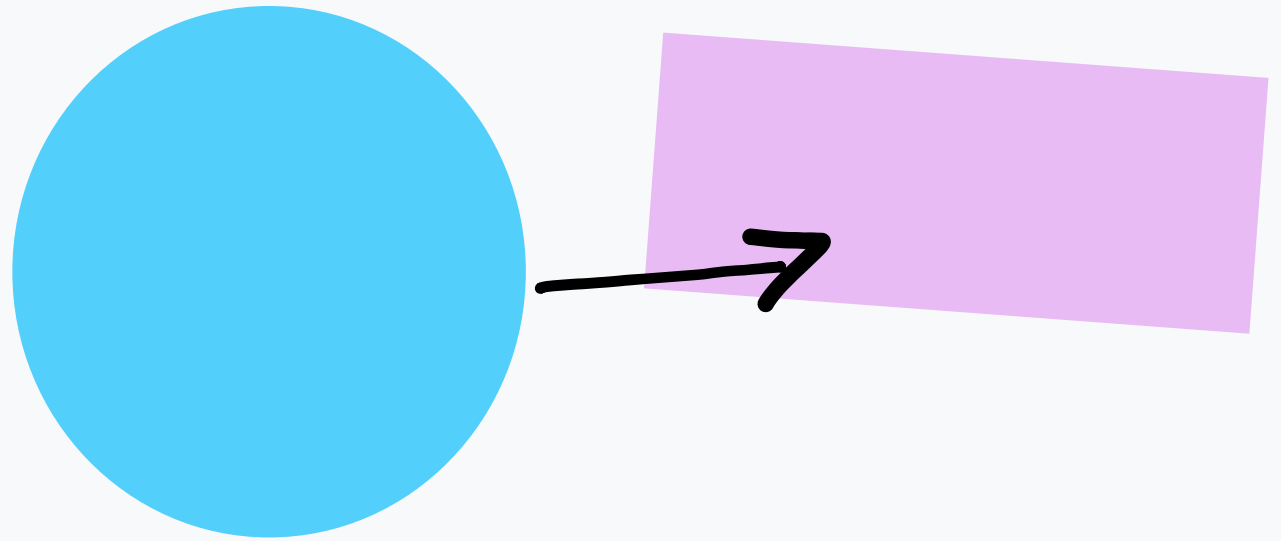
- Tamás Csaba
- Kálóczi Dávid
- Szabó Gergely
- Vázsonyi Dániel
- Kovács Tibor

Agenda – Fenntartható Frontend Fejlesztés

- Mi az a UX?
- Kliens-Szerver és SPA
- Sebezhetőségek
- Fejlesztői eszközök
- Fenntartható fejlesztés
- Hasznos dolgok
- (valós példa minderre)

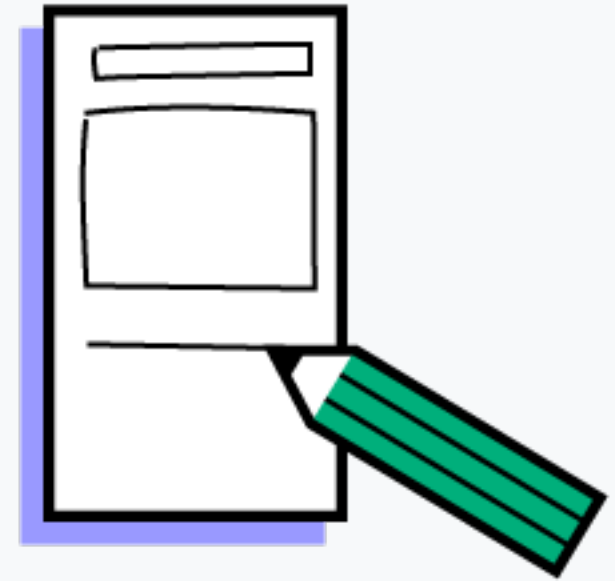
UX

A tervezés fontossága



UX – UI feladatok

- A jó felhasználói élmény része a kiszámíthatóság, megbízhatóság
- Egységes design, kiszámítható működés
- Az átgondolt funkciók kevésbé változnak fejlesztés során
- Kevesebb ”mi történik, ha...”



UX – Design Language

- Szabályrendszer, ami a vizuális kommunikációt definiálja
- Színek, árnyékok, formák választása amivel eszköztárat biztosítunk magunknak
- Komponensek, molekulák tervezése, ami az elemi interakciókért felelnek
- Mintázatok, amik gyakran előforduló, hasonló szerkezetekre adnak magyarázatot



Type

Title 1 · 44/56

Title 2 · 32/36

Title 3 · 24/28

Large · 19/24

Regular · 17/22

Small · 14/18

MICRO 1 · 8/8

A11y Color



Rausch
#FF5A5F
3.05:1



A11y Babu
#00A699
3.03:1



A11y Arches
#FC642D
3.0:1



A11y Hof
#484848
9.14:1



A11y Foggy
#767676
4.54:1

Spacing



8 · tiny



16 · small



24 · base



48 · large



64 · x-large

UX javítások

Jobb UX érdekében...

15:09 

[< Back](#)

6.35.0

2w ago

This version contains bug fixes, new functionalities, and various UX improvements. Happy shopping!

6.34.1

1mo ago

This version contains bug fixes, new functionalities, and various UX improvements. Happy shopping!

6.34.0

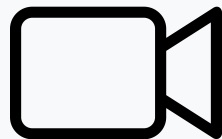
1mo ago

In this version, we have fixed bugs and added new functionalities and various UX improvements.

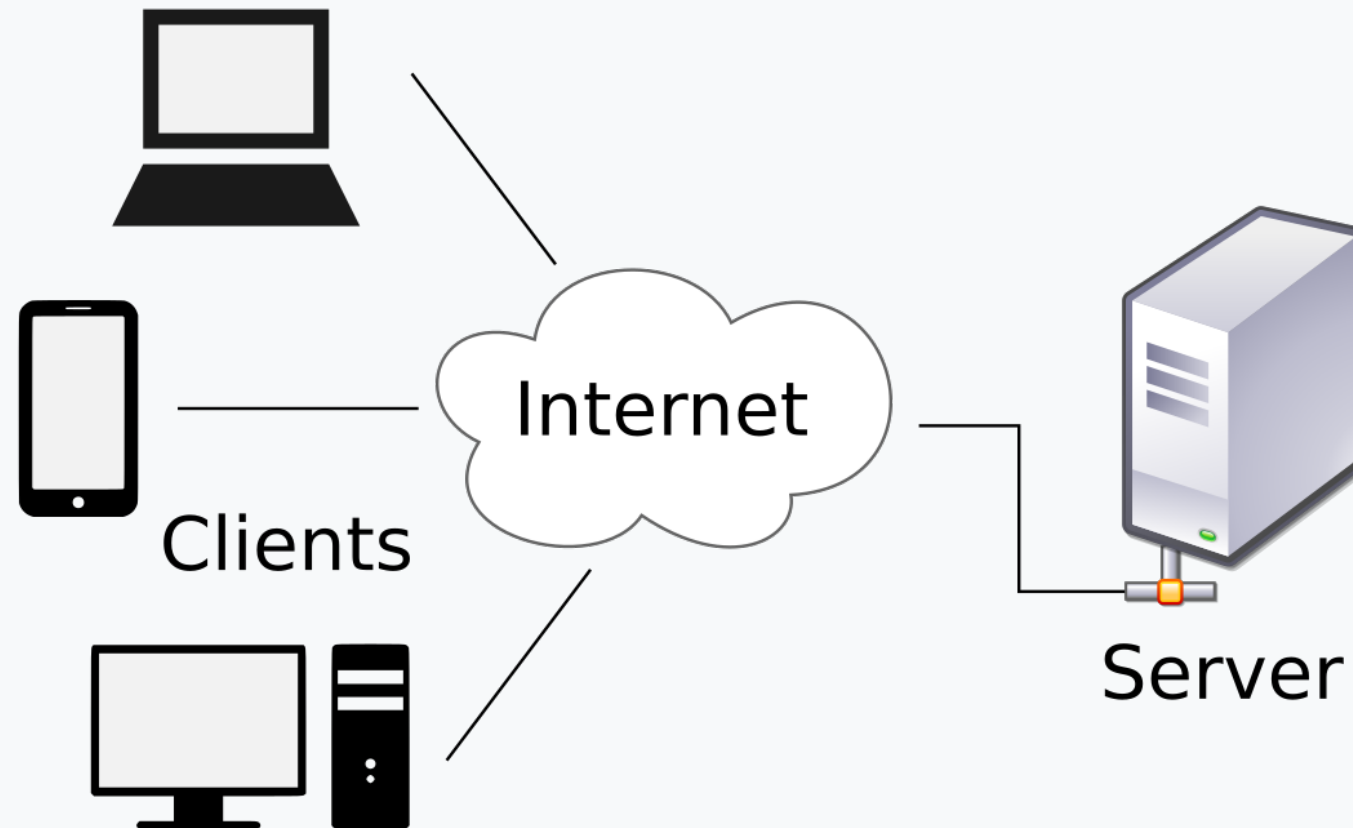
A kliens-szerver architektúra

Intro - web

- Egy olyan *service* ami lehetővé teszi az adatmegosztást különböző eszközök között
- Az adat lehet

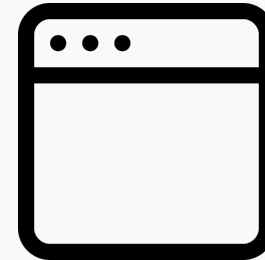
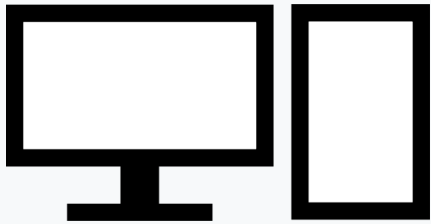


Intro – kliens-szerver



Kliens

- Valamilyen eszköz - client machine - vagy program – client program
- Eszköz - appliance - és egy mód arra, hogy *requesteket* küldjünk és fogadjunk



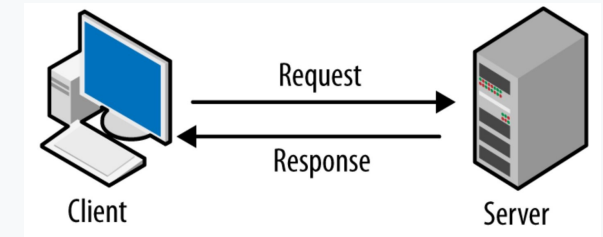
Szerver

- A szerver egy program, **nem maga a fizikai gép**
- Nagyteljesítményű gép > szerver > valamilyen *programot* futtatnak
- Valamilyen funkcionalitást nyújtanak és kiszolgálnak más programokat – a klienseket
- Több klienst - párhuzamosan
- Egy szervergép futtathat több szerver programot – virtualis szerverek

express



- Hosztolhat web appokat, tarolhat adatot, egyéb fajlokat, programokat, etc.
- Szamos kerest teljesit egyszerre – több ezer
- Mukodesi elve: a request – response cycle via HTTP(S)
- A kliens-szerver csak egy lehetsleges opcio - centralizalt
 - Serverless
 - Peer-to-peer - decentralizalt



Single Page Applications

SPA vs Traditional

- JavaScript elonyeire epit
- Kisebb szerver terheles
- Kenyelmesebb es gyorsabb hasznalhatosag
- Kod ujrahasznalhatosag – komponens alapu fejlesztés
- Facebook, Twitter, Gmail

Initial load

Traditional

Single Page
Application

Data change

Traditional

Single Page
Application

Elonyok es hatranyok

- Gyorsabb fejlesztés – szeparalt frontend es backend
 - Backend /API/ ujrafelhasznalas – mobil verzio
 - Jobb UX
-
- Rossz SEO
 - Sebezhetosegek "gyakoribbak"

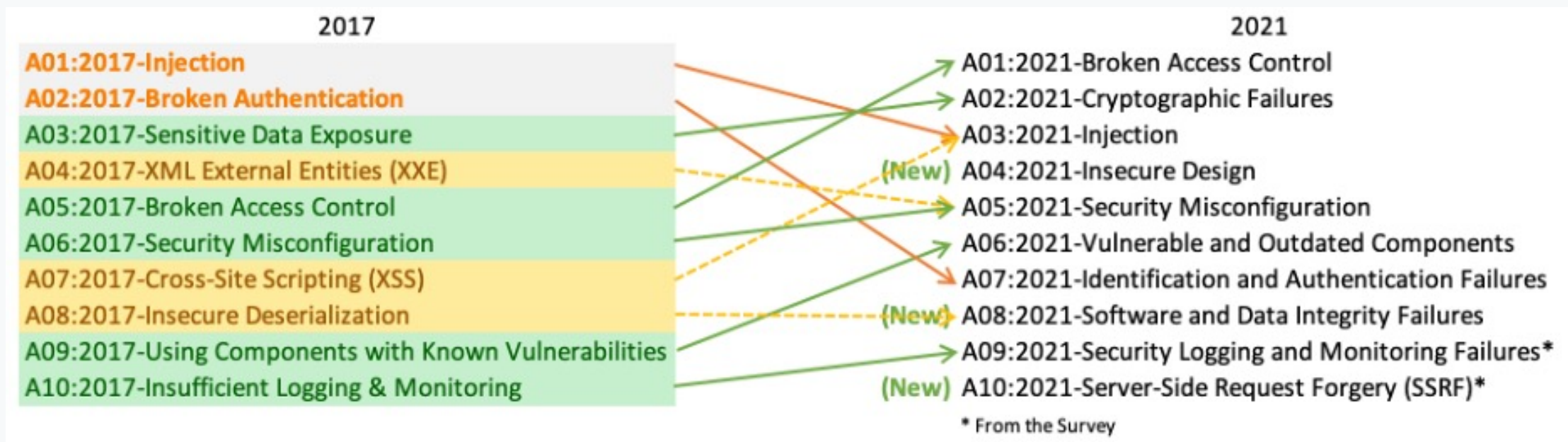
Sebezhetőségek

A legjobb védekezés a támadás

OWASP TOP 10

Open Web Application Security Project

<https://owasp.org/www-project-top-ten/>



SQL injection

- Egyik legnépszerűbb webes hackelési technika
- A weboldal beviteli mezőjén keresztül, rosszindulatú kód elhelyezése SQL utasításokban
- Az SQL támadás céljai
 - Szenzitív adatokhoz való hozzáférés
 - Rendszer hiba előidézése amit más támadásokhoz segítenek hozzá
 - Rendszer/adatbázis tönkretétele

Általános SQL injection-befecskendezés

1. Bejelentkezési felület

2. Szerver oldali feldolgozás

Username:

Password:

John Doe/myPass



`SELECT * FROM Users WHERE username = '<username>' AND password = '<password>'`

`John Doe ' OR 1=1; --`

`SELECT * FROM Users WHERE username = "' OR 1=1; --<username>' AND password = '<password>'`

`'; DROP TABLE Users; --`

`SELECT * FROM Users WHERE username = "'; DROP TABLE Users; --<username>'`

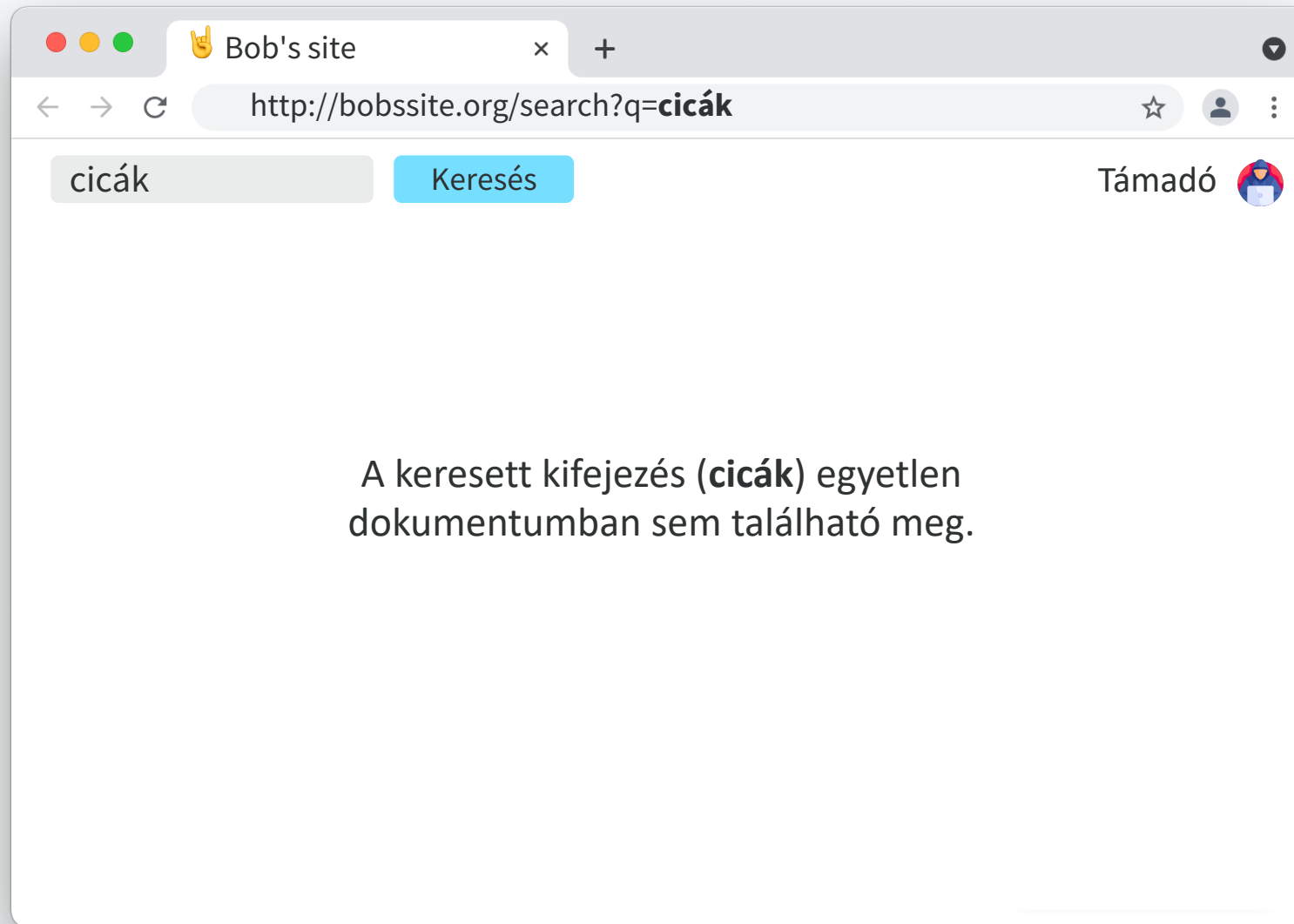
`AND password = '<password>'`

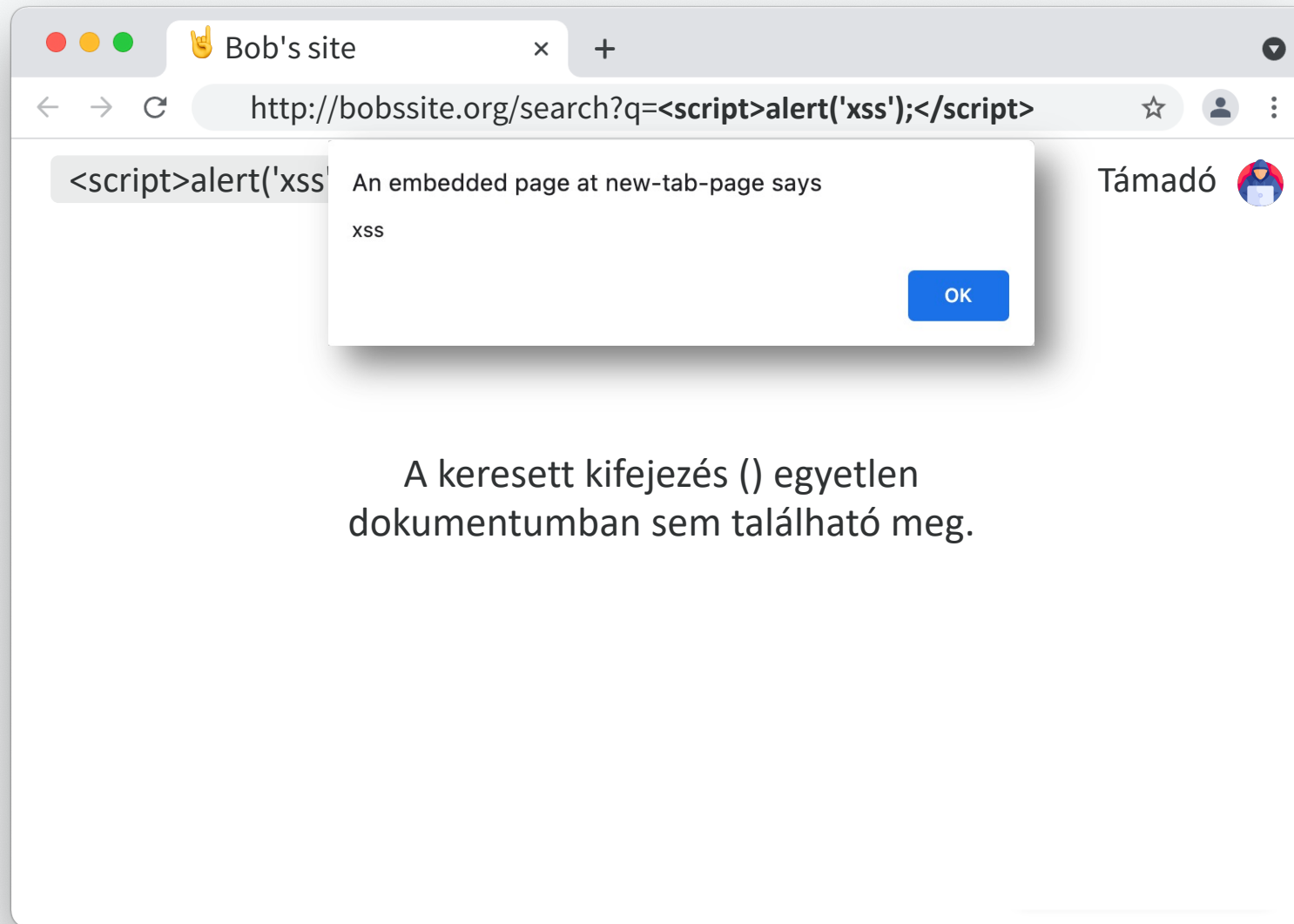


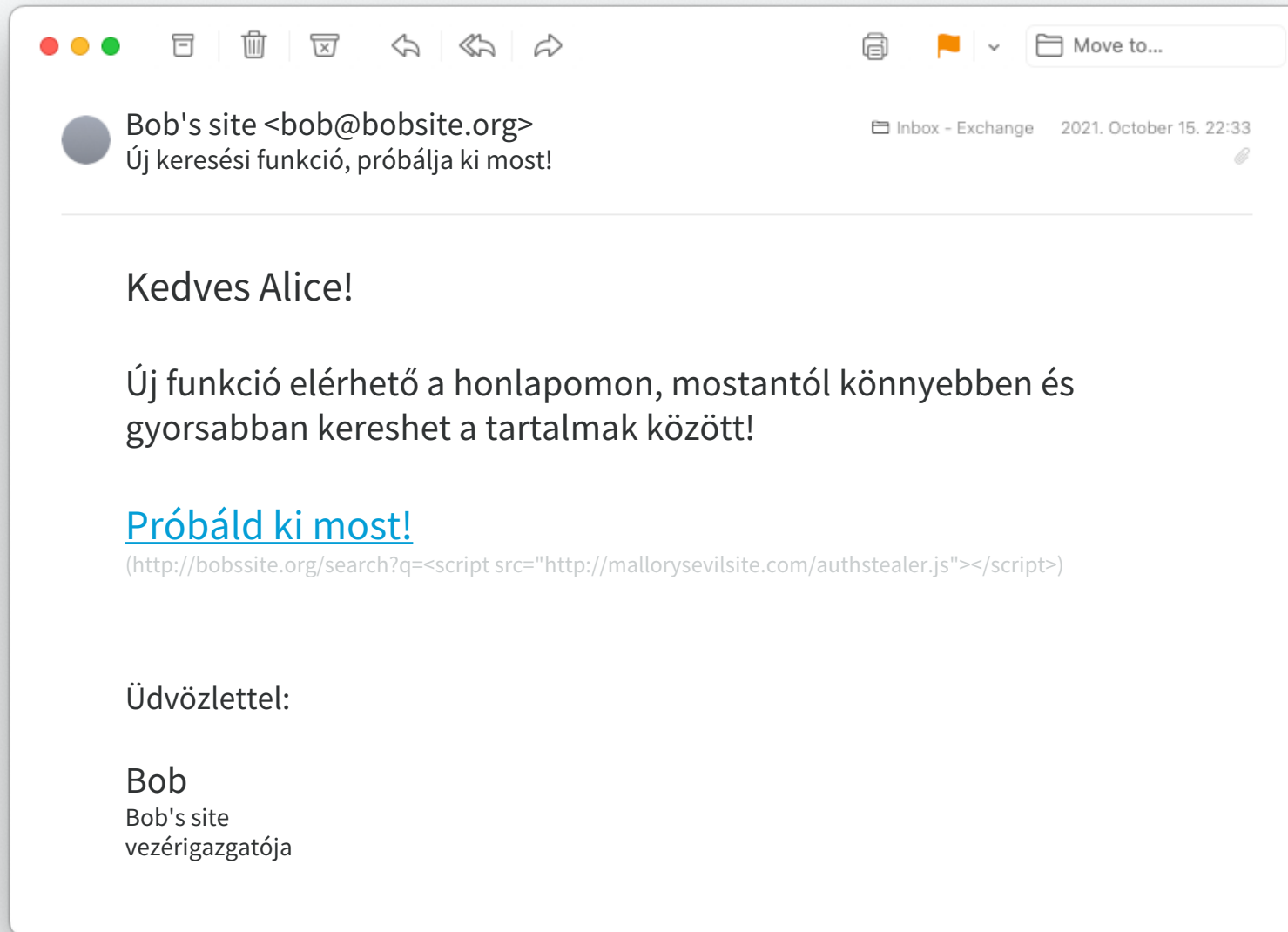
Cross-Site Scripting (XSS) Attack

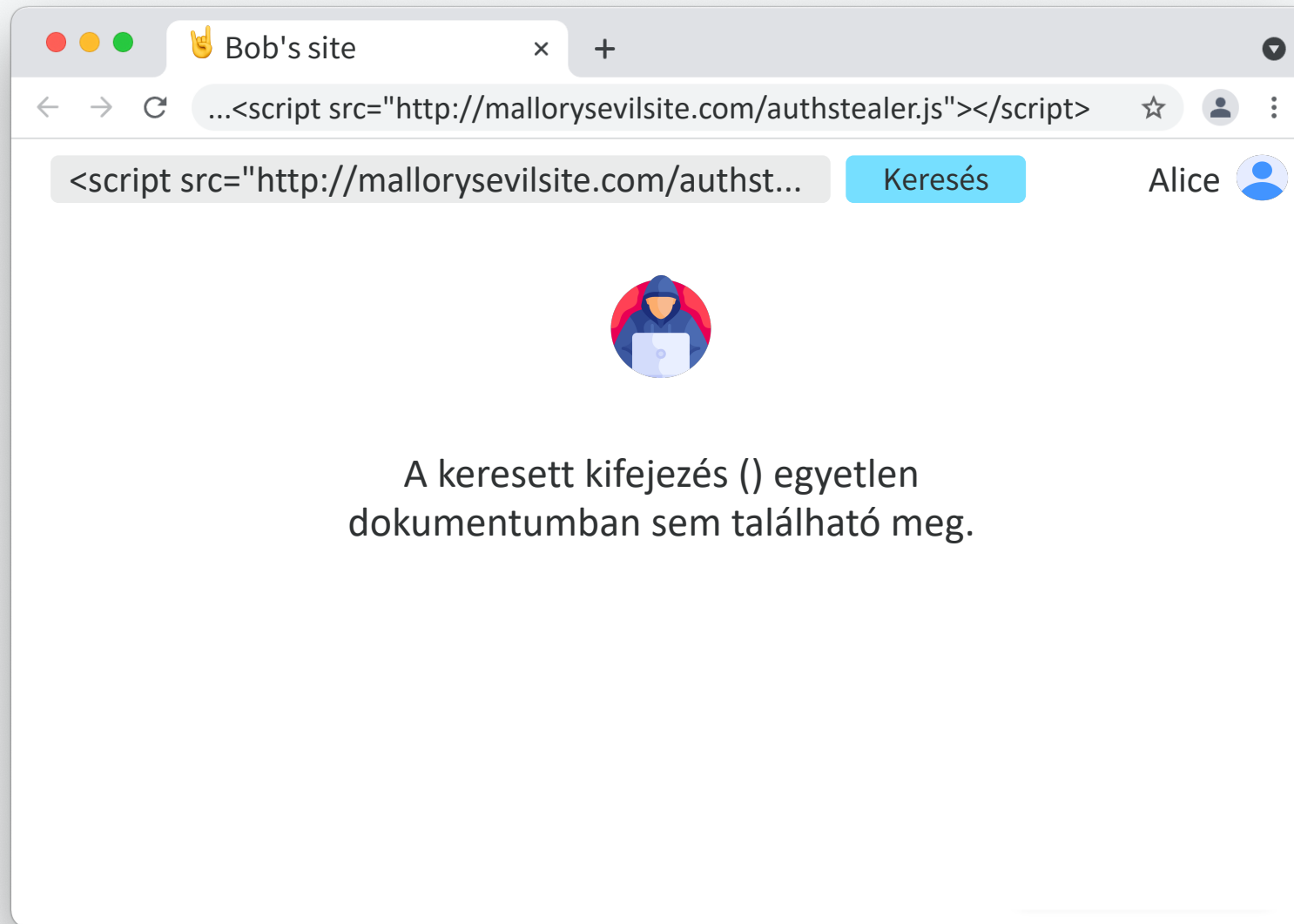
- Kihasználja a szerveret az oldalátogatók megtámadására
- A támadás célpontja nem a szerver, hanem a felhasználók
- Itt a támadás lehet elmentett/állandó vagy ideiglenes

Átmeneti XSS támadás

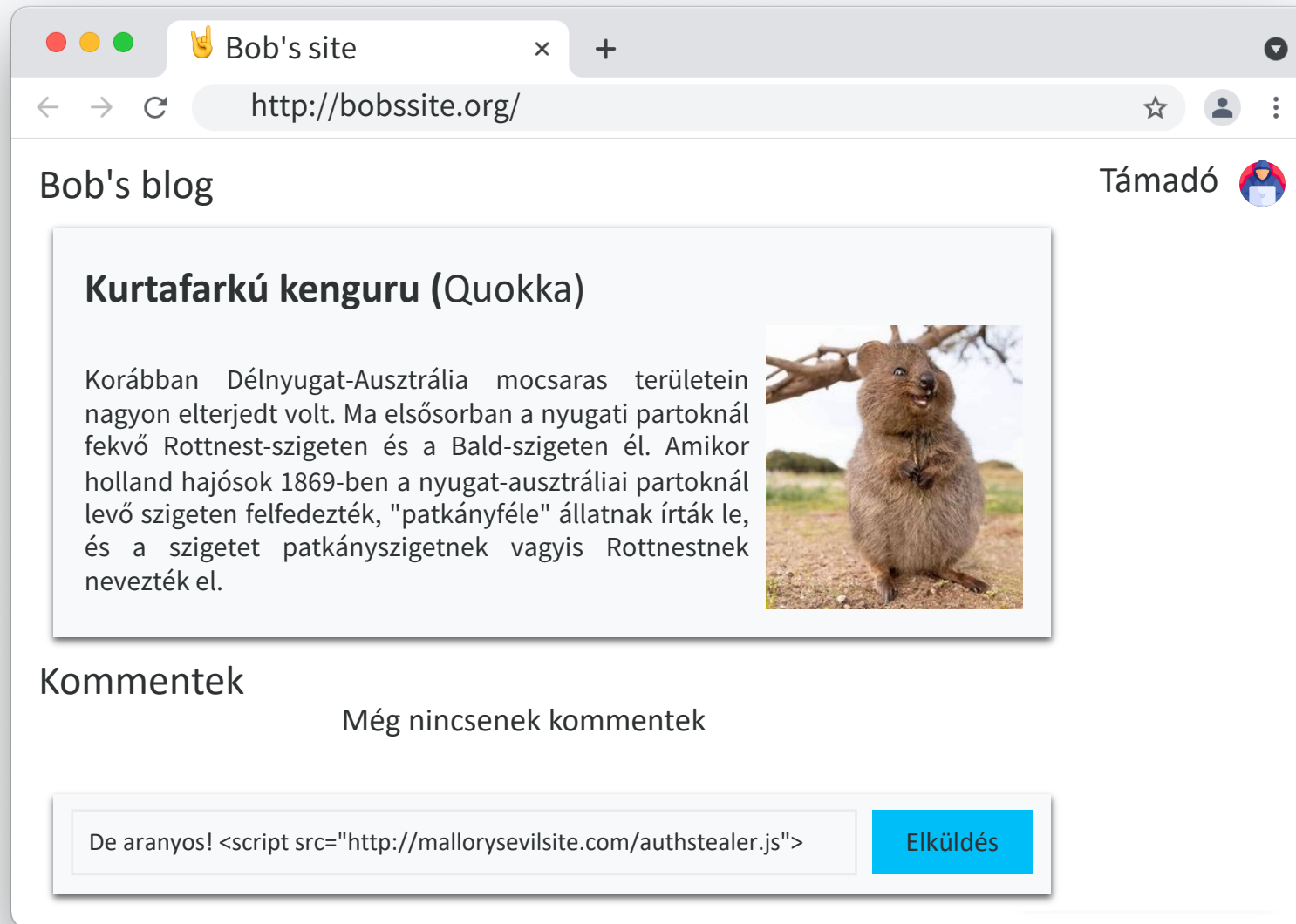


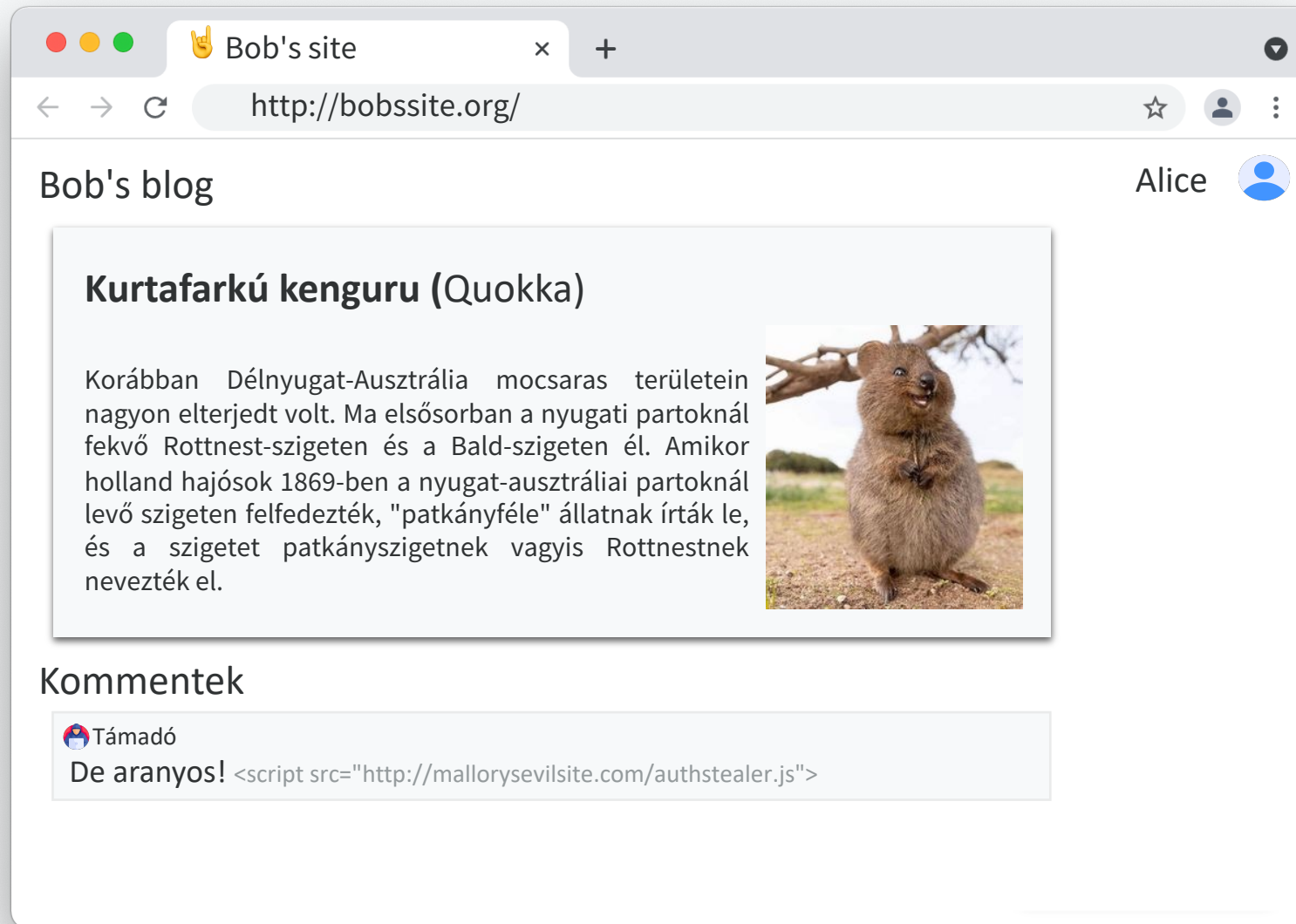






Elmentett/állandó XSS támadás



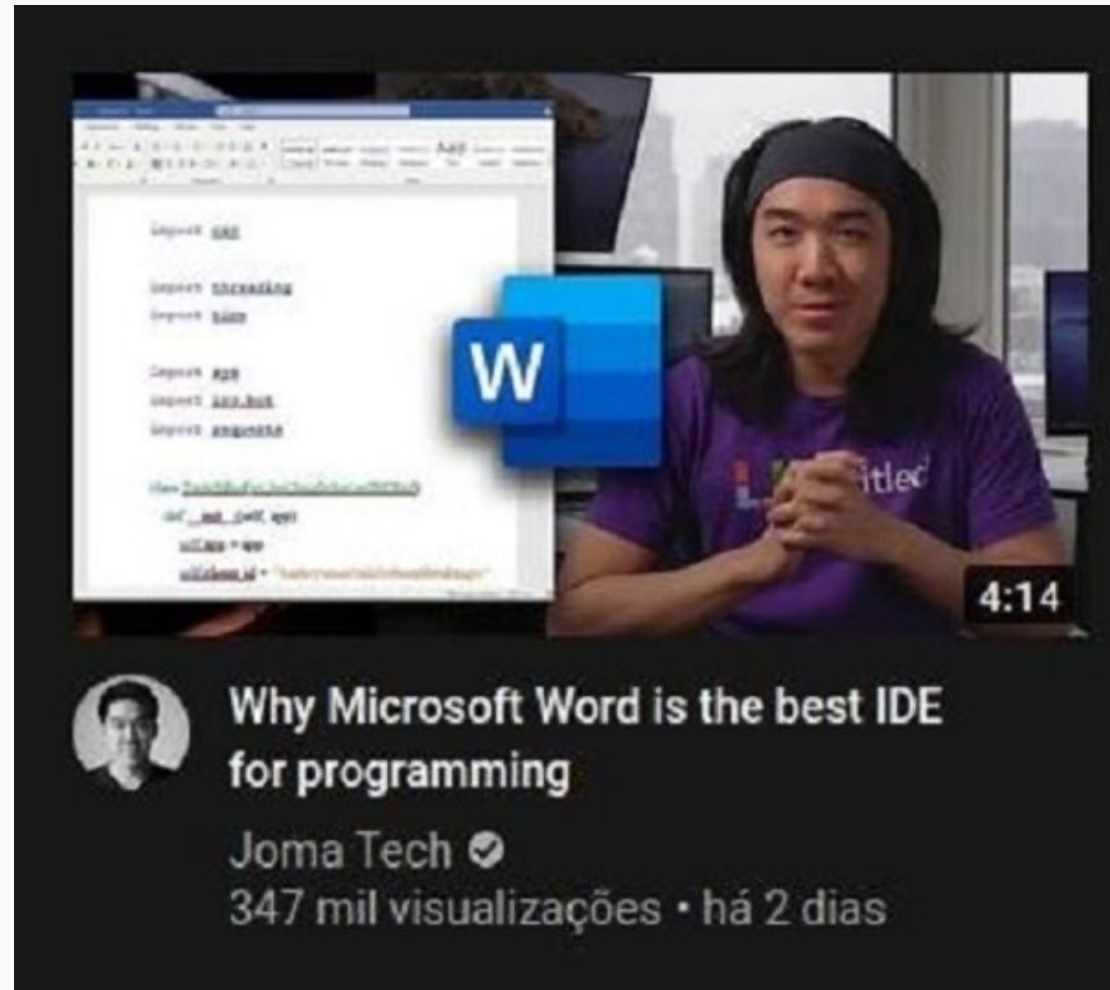


Megelőzés

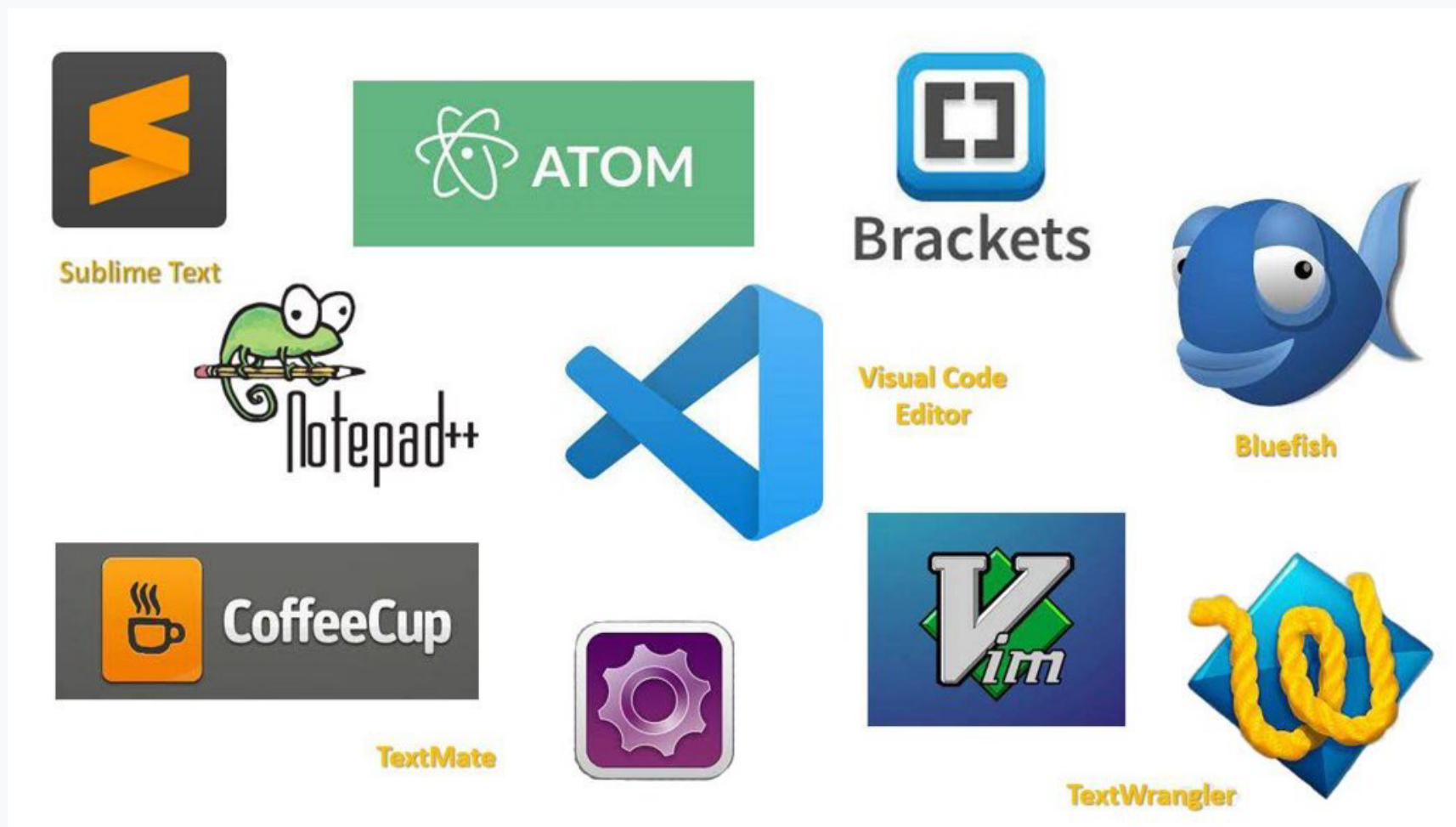
- Legjobb védekezés a támadás
- Automatikus SQL Injection és adatbázis feltérképező eszköz
<https://sqlmap.org/>
- Minden technológia/nyelv más-más sebezhetőséggel rendelkezik
- OWASP Puska és legjobb megoldások forráskódja
<https://github.com/OWASP/CheatSheetSeries/>

Fejlesztői eszközök

Legjobb editorok a webfejlesztéshez



Thousands of text editors



VSCode vs Webstorm



Postman and Hoppscotch



Wallaby



Browsers

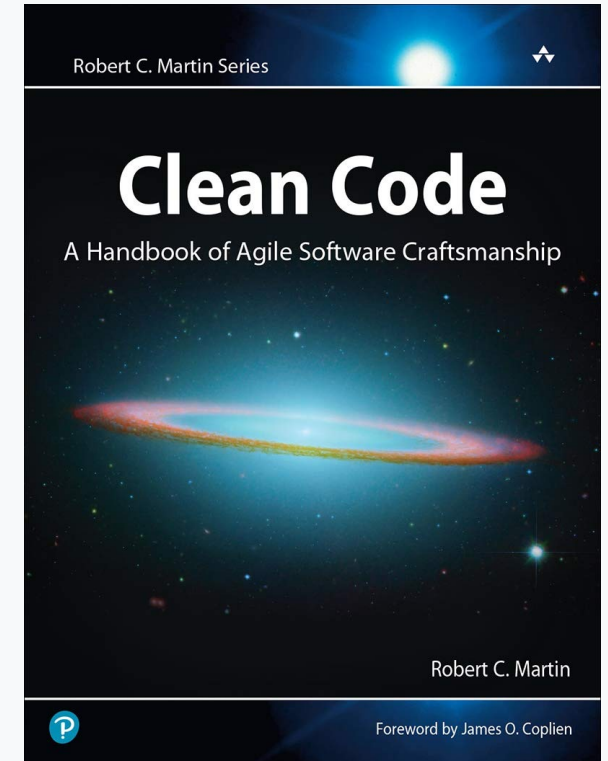


Fenntartható Fejlesztés

Clean code, lint, testing, CI/CD

Clean code

- A tiszta kód egy olyan fejlesztő számára is olvasható, továbbfejleszthető, aki nem az eredeti szerző
- Értelmes elnevezések vannak benne (angolul)
- A logikája egyszerű
- Egy dolog elvégzéséhez egy módot biztosít.
- Minimális függőségei vannak
- További kulcsszavak: SOLID, DRY,



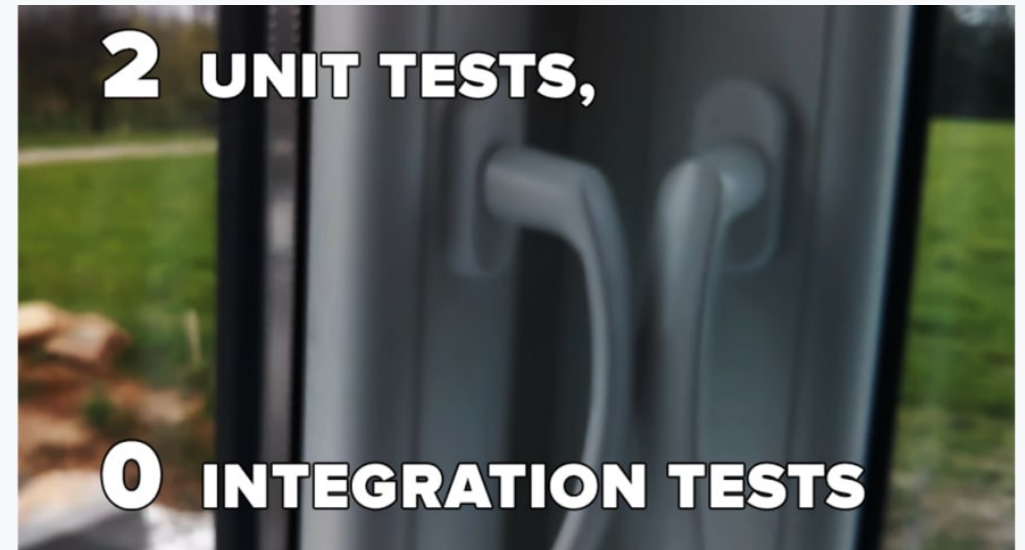
Lint

- Egy statikus kódelemző eszköz
- Segít kiszűrni a programozási hibákat
- Sok esetben ki is tudja javítani (--fix)
- Legelterjedtebb linter: eslint
- Lint mellett a kódformázás is fontos:
 - Egységes kinézet segíti az olvashatóságot
 - Beépül a legtöbb editor-ba/ide-be
 - Prettier a leginkább elterjedt



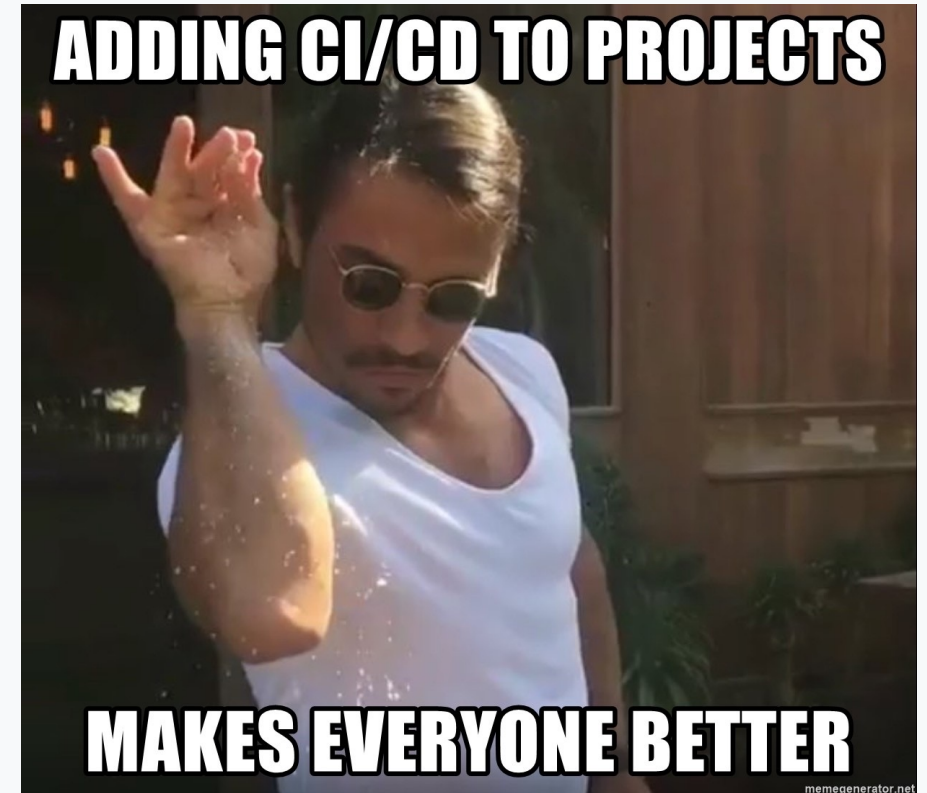
Testing

- A tesztelés egy rendszer vagy program kontrollált körülmények melletti futtatása, és az eredmények kiértékelése
- Többféle tesztelési mechanizmust különböztetünk meg:
 - Unit
 - Integration
 - End to end (e2e)
 - Visual



CI (Continuous Integration)

- A kód bekerülésének (integrálásának) automatizálását fedi le.
- Kezeli a tesztek, lintek és egyéb fontos tool-ok automatikus futtatását változás esetén.
- Nagybán segíti a csapatmunkát
- Ismertebb rendszerek:
 - Jenkins, travis, CircleCI,



Tools

- License checker
- Security checker
- Copyright checker
- Dependency health checker
- Performance checker
- A11y checker
- SEO checker



Hasznos linkek

<https://veszprem.dev/fenntarthato-frontend>