

Budapest Műszaki Szakképzési Centrum

Verebély László Technikum

54 213 05 Szoftverfejlesztő

tanfolyam

NextCaligo

Linux szerver kezelő szoftver

Készítette: Stütz Attila Erik

Konzulens: Horváth Attila

2022/23-as tanév

## Tartalom

Bevezetés .....	4
Cél .....	5
Fejlesztői környezet .....	6
Futtatási környezet .....	6
Adatbázis tervezet .....	7
Adatbázis táblaszerkezete .....	8
Felhasználói felület .....	13
Minőségi követelmények .....	14
Felhasznált könyvtárak .....	15
Terjesztés .....	16
Fejlesztési ütemterv .....	16
Futtatási környezet kialakítása .....	16
Adatbázis megtervezése .....	16
Szoftverfejlesztés .....	17
Tesztesetek .....	17
Funkcionalitási terv .....	19
Rendszerszereplők .....	19
Folyamatábra <sup>1</sup> .....	20
Implementációs terv .....	21
Fiókok kezelése .....	21
Távoli szerverek kezelése .....	22
Virtuális szerverek .....	23
Fizikai környezet .....	25
Felhasználói dokumentáció .....	26
Rendszerkövetelmények .....	26
A program bemutatása .....	26

Telepítési útmutató.....	28
Használat.....	29
Bejelentkezés .....	29
Elfelejtett jelszó .....	30
Vezérlőpult.....	30
Szerverek menü.....	30
Szerver menedzsment .....	30
Szerverek hozzáadása .....	32
Virtuális szerverek .....	33
Virtuális szerver létrehozása .....	34
Virtuális szerver kezelése .....	34
Tűzfal & biztonság.....	35
Tevékenységek.....	36
Felhasználók .....	36
Felhasználó kezelése .....	37
Felhasználó hozzáadása .....	37
Központi szerver menedzsmentje .....	38
Erőforrás & statisztika .....	39
Saját fiókom.....	39
Hibakezelés .....	40
Összegzés.....	41
Fejlesztési lehetőségek.....	42
Köszönetnyilvánítás .....	43
Források .....	44
Ábrajegyzék .....	44

# Bevezetés

A szakdolgozatom a NextCaligo névre hallgat, a „Next”, mint „következő” angol szóból, a „Caligo” pedig a latin „kőd” szóból ered, ezzel utalva a legfőbb specializációmra és érdeklődési körömrre, a felhő alapú technológiákra.

Érdeklődési körömnnek megfelelően a szoftver egy olyan webes felületet képez, amely egyetlen központi grafikus vezérlőpultból képes menedzselni a nyilvántartott Linux disztribúción alapuló távoli szervergépeket – rendszergazdai jogkörrel rendelkezve képes azokat leállítani, újraindítani, megváltoztatni a számítógép nevét, megváltoztatni az adminisztrátori fiók jelszavát, megtekinteni a háttérben futó folyamatokat és megtekinteni a processzor, a memória és a merevlemez erőforrás teljesítményét. A szoftver lehetőséget biztosít a rendszergazdai teendők kényelmes elvégzésére egy webes felületről.

A szoftver megkülönbözteti a felhasználókat kompetenciájuk és feladataik szerint, végfelhasználói és rendszergazdai jogköröket biztosít. A rendszergazdai jogosultsággal bírók a bejelentkezést követően képesek menedzselni, komplex műveleteket végrehajtani a rendszerhez hozzáadott Linux szervereken, a végfelhasználók pedig kizárólag a szerverek erőforrás statisztikáit képesek monitorozni. A szoftver jellegéből adódóan a szoftver nem használ nyílt regisztrációt, a rendszergazda jogosult az általa megszabott jogkörrel fiókokat hozzáadni a rendszerhez.

A szoftver létrehozása során nagy hatással volt rám a témában a szakdolgozatommal hasonló funkciókkal bíró, és méltán híres SolusVM alkalmazás, amely távoli szervergépek, kiszolgálók összekötésével valósítja meg a rendszerkezelést, illetve informatikai szolgáltatások széles palettáját kínálja a használóinak és ügyfélkörének: Linux alapú virtuális szerverek létrehozása egy központi felületről, amely szolgáltatás a világ bármely pontján elérhető, a szolgáltatás a tetszőleges országba kerül elhelyezésre.

Eddigi meglátásaim –, a személyes környezetemből kiindult tapasztalatok – szerint a közép- és nagyvállalatok a nagy mennyiségű adatok tárolására, feldolgozására és a szolgáltatásaik létrehozására, illetve menedzsmentjére felhő alapú, avagy egyszerre több szerverre épülő technológiákat használnak, amelyet egyetlen központi, akár webes felületről a vállalat felhasználói könnyedén elérnek. Ebben a technológiában és struktúrában látom a jövő vállalati megoldásait: hitem szerint minden a felhőbe kerül majd.

# Fejlesztői dokumentáció

## Cél

A szoftver elsődleges célja, hogy egy többnyelvű, reszponzív PHP alapú platformot biztosítson az informatikában dolgozóknak, főként a rendszergazdáknak, adatközponti technikusoknak, fejlesztőknek, és minden olyan munkakörben lévő személynek, akinek egyszerre több szerver szükséges a projektjei kényelmes kezelésére.

A bejelentkezést követően a felhasználó jogkörének megfelelő funkciókat látja a felület oldalsó navigációs menüjében és csak azokat képes használni. A rendszergazdai jogkörrel rendelkező felhasználók képesek összetett műveletek végrehajtására: a helyi szerver és a távoli munkaállomások leállítása, újraindítása, root jelszó módosítása, számítógépnév (hostname) módosítása és a Linux rendszer processzor, memória és merevlemez erőforrás használatának megtekintése. Továbbá, a rendszergazdai jogkörrel felruházott felhasználók képesek virtuális szervereket létrehozni tetszőleges memória és tárhely mérettel, tetszőleges operációs rendszerrel, illetve képesek végfelhasználókat hozzáadni a rendszerhez és a fiókokat a kompetenciájuknak, feladatkörüknek megfelelő jogkörhöz hozzárendelni.

A végfelhasználók csak a szerverek erőforrás statisztikáit képesek megtekinteni, illetve minden felhasználó jogkörétől függetlenül lehetősége nyílik a saját fiókjának a kezelésére – képes jelszót és nevet változtatni, megnézni a fiókjával kapcsolatos naplózott eseményeket, illetve igény esetén a felhasználói fiókot felfüggeszteni.

A szoftver része egy webes tűzfal is, amely kísérletet tesz a Tor böngészőből, illetve VPN kapcsolattal rendelkező látogatók blokkolására, ugyanis ezen technológiák lehetővé teszik a látogatók IP címének elrejtését, így a tevékenységük anonimitását is, amivel a naplófájlokba nem a látogatók valós IP címe kerül.

# Fejlesztői környezet

Kódszerkesztésre és annak átláthatóságára Visual Studio Code-ot használtam a szoftver fejlesztése során, amely az SSH2 kiegészítő funkciójával nagyban hozzájárult ahhoz, hogy a fájlokat az otthoni asztali számítógépről közvetlen a szakdolgozat kidolgozására üzembehelyezett szerveren tudjam szerkeszteni, tárolni. Ezen bérszerver egy Ubuntu 22.04 operációs rendszert szolgál ki, amelyen OpenSSH, Apache2, PHP 8.1, MariaDB 8 – és az adatbázis eléréséhez használt PhPMyAdmin – foglal helyet.

A projekt elkészítéséhez azért az Ubuntu volt a legideálisabb számomra, mert fejlesztőbarát eszközökkel rendelkezik, nagyon könnyen telepíthető és használható rajta az NPM szoftvercsomag kezelő.

A fejlesztői környezet kialakítására egy virtuális bérszervert használtam, ugyanis képes egy valós szervergép környezetét emulálni, illetve annak erőforrásait úgy kezelni, mintha fizikai munkaállomáson működne a szoftver. A szerver 2 GB RAM memória, 50 GB SSD tároló és 1 CPU erőforrást használ a szoftver kiszolgálásához a fejlesztés és tesztelés ideje alatt.

# Futtatási környezet

A szoftver elérését Apache2 webszerver biztosítja, amelyhez HTACCESS hozzáférés van bekonfigurálva, annak érdekében, hogy a PHP alapú fájlok módosított URL címeken legyenek elérhetők a böngészőből. A platform bármely böngészőből hozzáférhető kliens oldalról, amely képes JavaScript futtatására.

Szerver oldalról az alkalmazás számítási, „backend” feladatait a PHP 8.1 látja el a következő kiegészítőkkal:

- cli,
- mysqli,
- mbstring,
- bcmath,
- gd,
- gettext,
- php-ssh2

A PHP-SSH2 modul lehetőséget ad arra, hogy PHP kóddal lehessen SSH parancsokat kiadni távoli szervereknek, anélkül, hogy közvetlen OpenSSH terminál parancsokat kellene kiadni a kapcsolódáshoz.

Az adattárolást és adatbázis műveleteket a MariaDB adatbázis szerver látja el. Választásom azért erre az adatbázis szerverre esett, mert szerkezetében kísértetiesen hasonlít a hagyományos MySQL adatbázisra, mégis biztonságosabb és gyorsabb adatforgalmat képes lebonyolítani.

OpenSSH eléréssel az adatforgalmat fogadó szervernek kell rendelkeznie, azaz a rendszerhez hozzáadott, elérni kívánt szervernek, amelyet vezérelni szeretnénk a szoftver irányítópultjából. Ezeknek a távoli szervereknek az optimális működés érdekében Ubuntu vagy Debian alapú operációs rendszerrel kell működniük, ugyanis ezen disztribúciókon lettek tesztelve azok a rendszerparancsok, amelyeket a szoftver elküld a vezérelni kívánt szervernek, majd a kiadott parancs lefutás után ezekből a válaszokból állítja össze a felhasználói felületen megjelenő eredményeket.

A virtuális szerverek létrehozásához KVM kernel-szintű virtualizációt építettem a szoftverbe, amelynek a Libvirt segédeszköz szoftvercsomag használatával parancssoros formában lehet virtuális szervereket létrehozni, illetve paraméterezni azokat nevük vagy azonosítjuk alapján: lehetséges őket tetszőleges operációs rendszer ISO-val elindítani, konfigurálni a memória és a tárhely méretét, beállítani a processzormagok számát. A virtualizációs rendszerműveleteket a PHP `shell_exec` függvénye teszi lehetővé, illetve az ahhoz fejlesztett KVM névre hallgató objektumosztály.

## Adatbázis tervezet

Az adatbázis célja a felhasználói fiókok adatainak eltárolása, illetve a fiókok tevékenységeinek rögzítése, naplózása egy különálló táblába, továbbá a sikeres bejelentkezések körülményeinek rögzítése. A távoli szerverek SSH2 bejelentkezési adatai szintén egy táblában kerülnek eltárolásra és az ahhoz tartozó tevékenységek naplózása is külön táblában történik. A virtuális szerverek alapvető erőforrás paraméterei is egy táblában tárolódnak, és a virtuális szerverekhez tartozó biztonsági mentések nevei és a pontos dátumai is.

Minden gomb lenyomása, amely olyan módosításokat végez a Linux rendszeren vagy a felhasználói fiókon az naplózásra kerül, új sorokat rögzít az adatbázisban a tevékenységet

végző felhasználó adataival: felhasználóneve, IP címe, a cselekmény pontos leírása és a véglegesített cselekmény pontos ideje. A fokozott naplózás célja, hogy minden tevékenység visszanezhető és ellenőrizhető legyen.

Az adatbázis elérése érdekében PHP oldalról egy SQL könyvtárat használtam fel, amely gyors és biztonságos kapcsolatot létesít lekérdezések, frissítések és implementációk esetén, ezzel védekezve a MySQL injection támadások ellen. A fájl a CodeShack weboldaltól került felhasználásra, amely kód nyílt forráskódú és tulajdonosa a honlap felhasználási feltételeiben engedélyezte a használatot, módosítást és a terjesztést is.

## Adatbázis táblaszerkezete

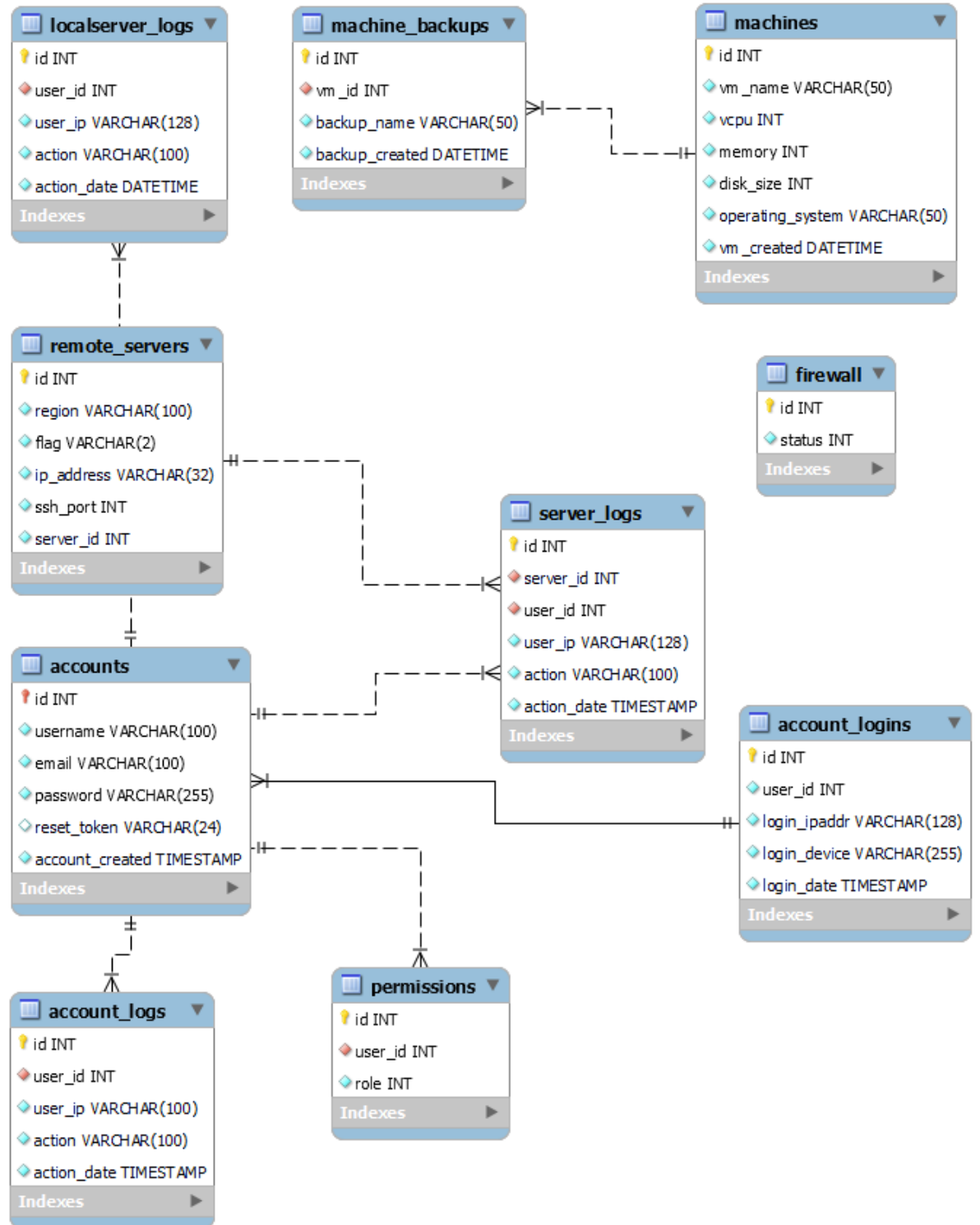
- Az „accounts” tábla a felhasználók bejelentkezési adatait tárolja, biztosítja a jelszavas bejelentkezést és az elfelejtett jelszó emlékeztetéséhez használt generált kódot.
  - id: integer típusú elsődleges kulcs;
  - username: 100 karakter hosszú szöveges mező, amely a PHP-Session alapú bejelentkezést teszi lehetővé, így szoftveresen biztosított, hogy az értéke ne ismétlődhessen;
  - email: 100 karakter hosszú szöveges mező;
  - password: 255 karakter hosszú szöveges mező, a jelszó Argon2ID algoritmus titkosított, visszafejthetetlen formáját tárolja;
  - reset\_token: A mező alapértelmezett értéke NULL. Amennyiben a felhasználó kódot generál fiókja hitelesítése érdekében, akkor 24 hosszúságú véletlenszerű betűkből és számokból álló kód jelenik meg benne, melynek egyszeri használata után ismét NULL értéket vesz fel.
  - account\_created: datetime típusú érték, a felhasználói fiók létrehozásának időpontját tárolja.
- A „permissions” tábla tárolja a felhasználók jogait. Az alkalmazásban csak felhasználói vagy rendszergazdai fiók létrehozása lehetséges.
  - id: integer típusú elsődleges kulcs;
  - user\_id: integer típusú idegen kulcs, a felhasználó azonosítóját tartalmazza;
  - role: integer típusú mező, hossza egy karakter, csak 0 vagy 1 értékű lehet, alapértelmezett értéke nulla. A 0 érték jelöli az általános felhasználói jogosultságot, az 1 pedig az adminisztrátori hozzáférést.



- Az „account\_logins” tábla a sikeres bejelentkezési adatokat tárolja.
  - id: integer típusú elsődleges kulcs;
  - user\_id: integer típusú idegen kulcs, a felhasználó azonosítóját tartalmazza;
  - login\_ipaddr: a felhasználó IP címét tartalmazza, 128 karakter hosszúságú szöveges mező;
  - login\_device: Az eszköz adatait tartalmazza, amelyről a bejelentkezés megtörtént, 255 karakter hosszúságú szöveges mező;
  - login\_date: datetime típusú érték, timestamp jelenlegi időpillanatot tárolja
- Az „account\_logs” tábla a már bejelentkezett felhasználók tevékenységeit naplózza.
  - id: integer típusú elsődleges kulcs;
  - user\_id: integer típusú idegen kulcs, a felhasználó azonosítóját tartalmazza;
  - user\_ip: a felhasználó IP címét tartalmazza, 128 karakter hosszúságú szöveges mező;
  - action: a felhasználó tevékenységét részletező mező, 100 karakter hosszú szöveges mező;
  - action\_date: datetime típusú érték, a bejelentkezés pontos dátumát tárolja.
- A „localserver\_logs” tábla a helyi, központi szerveren végzett módosítások tevékenységeit naplózza.
  - id: integer típusú elsődleges kulcs;
  - user\_id: integer típusú idegen kulcs, a felhasználó azonosítóját tartalmazza;
  - user\_ip: a felhasználó IP címét tartalmazza, 128 karakter hosszúságú szöveges mező;
  - action: a felhasználó tevékenységét részletező mező, 100 karakter hosszúságú szöveges mező;
  - action\_date: datetime típusú érték, a cselekmény pontos dátumát tárolja.
- A rendszerhez hozzáadott távoli szerverek a „remote\_servers” táblában kerülnek eltárolásra.
  - id: integer típusú elsődleges kulcs;
  - region: 100 karakter hosszú szöveges mező;
  - flag: két karakter hosszú szöveges mező, a megadott ország zászlókódját tartalmazza;
  - ip\_address: a szerver IP címét tartalmazza, 32 karakter hosszúságú szöveges mező;

- ssh\_port: integer típusú, 4 karakter hosszú mező;
- server\_id: Integer típusú, 6 karakter hosszú számkód, a Linux rendszerben a szerver SSH kulcsának nevére utaló véletlenszerűen generált kód.
- A „server\_logs” tábla a távoli szervereken végzett módosításokat naplózza.
  - id: integer típusú elsődleges kulcs;
  - user\_id: integer típusú idegen kulcs, a felhasználó azonosítóját tartalmazza;
  - server\_id: integer típusú idegen kulcs, a szerver azonosítóját tartalmazza;
  - user\_ip: a felhasználó IP címét tartalmazza, 128 karakter hosszúságú szöveges mező;
  - action: a felhasználó tevékenységét részletező mező, 100 karakter hosszú szöveges mező;
  - action\_date: datetime típusú érték, a cselekmény pontos dátumát tárolja.
- A „machines” táblában kerülnek eltárolásra a létrehozott KVM virtuális szerverek adatai.
  - id: integer típusú elsődleges kulcs;
  - vm\_name: 50 karakter hosszú szöveges mező;
  - vcpu: integer típusú mező, a virtuális szerver processzormagjainak számát jelöli;
  - memory: integer típusú mező, a virtuális szerver memóriáját tárolja MB mértékegységben;
  - disk\_size: integer típusú mező, a virtuális szerver merevlemez méretének eltárolására szolgál, GB mértékegységben;
  - operating\_system: 100 karakter hosszú szöveges mező, az operációs rendszer csatolt ISO fájljának nevét tárolja el.
  - vm\_created: datetime típusú érték, a virtuális szerver létrehozásának időpillanatát tárolja.
- A virtuális szerverekről készült biztonsági mentések a „machine\_backups” táblában található. A tábla minden virtuális szerverhez hozzárendeli a biztonsági mentéseket nevük és időpontjuk alapján.
  - id: integer típusú elsődleges kulcs;
  - vm\_id: integer típusú idegen kulcs, a virtuális szerver egyedi azonosítóját tartalmazza;

- backup\_name: 50 karakter hosszú szöveges mező, a biztonsági mentés nevét tartalmazza;
  - backup\_created: datetime típusú érték, annak érdekében, hogy az esetleges visszaállítás esetében a felhasználó meghatározhassa, melyik időbeli rendszerállapothoz szeretné visszaállítani a szerverét.
- A „firewall” segéd tábla menti az opcionálisan be- és kikapcsolható webes tűzfal beállítását, nem áll logikai kapcsolatban semmilyen más táblával.
  - id: integer típusú elsődleges kulcs;
  - status: Egy karakter hosszú szám típusú mező, alapértelmezett értéke 0. Hossza 1 számjegy, 0 érték esetében a tűzfal kikapcsolt állapotban van, ha a mező értéke 1-re változik, akkor a tűzfal bekapcsolásra kerül, és a WebAppFirewall objektum osztály VPN kapcsolatokat blokkoló funkciója meghívásra kerül az alkalmazás minden oldalára.



1. ábra: SQL adatbázis táblák és logikai kapcsolataik

# Felhasználói felület

A grafikus felület elsődleges célja, hogy a Linux CLI parancsok alkalmazásához szakértelemmel nem rendelkező felhasználók is könnyedén tudják kezelni a szerverüket egy webes interfészről. A reszponzív felhasználói felület HTML5, CSS3 és JavaScript alapú, ezért minden webböngészőből elérhető, amely támogatja a JavaScript futtatását, illetve amennyiben nem Tor böngészőről van szó, ugyanis a szoftver megtagadja a hozzáférést a Tor böngészőt használó látogatóknak biztonsági okokból.

A szoftver kezdőlapja egy jelszavas bejelentkező felület, amely captcha kód megjelenítésével véd a robotok elleni támadásoktól, illetve az elfelejtett fiók jelszópótlására, megváltoztatására is automatizált módon lehetőség van: jelszó helyreállítási linket küld a szoftver a felhasználó email címére, amelyen új jelszót tud beállítani a felhasználó az adott fiókhoz.

A sikeres bejelentkezést követően megjelenik a felső navigációs menü, mely a felhasználó üdvözlésére hivatott, választhat magyar és angol nyelvű megjelenítések közül, továbbá kijelentkezhet felhasználói fiókjából. A bejelentkezés lehetővé teszi a főbb funkciók használatát, amely az oldalmenüben jelenik meg a felhasználók jogosultságainak megfelelően. Az általános felhasználói jogkörrel rendelkező felhasználók a vezérlőpulthoz, a szerverek listájához, az erőforrás statisztikához és a virtuális szerverek listájához férnek hozzá. Az adminisztrátori hozzáféréssel rendelkező felhasználók sokkal szélesebb jogkörrel rendelkeznek, képesek a szerverek menedzsmentjére. Az oldalmenüben hozzáférhető számukra – az általános felhasználói jogok mellett – a szerverek hozzáadása, a root jelszó váltás, a hostname (gépnév) váltás, az újraindítás, a tűzfal és biztonság, a tevékenységek és a felhasználók menüpont is elérhetővé válik. Ezen menüpontok lehetővé teszik a távoli szerverek hozzáadását, azoknak és a helyi szervergép újraindítását, az erőforrás statisztika megtekintését, root jelszó és hostname váltást, a webes tűzfal be- és kikapcsolását, a virtuális szerverek létrehozását és menedzsmentjét, a felhasználók tevékenységeinek visszatekintését és a felhasználók adatlapjának szerkesztését, jogainak megváltoztatását. Mindezek mellett a bejelentkezett felhasználók jogkörüktől függetlenül képesek megtekinteni a legutóbbi bejelentkezések adatait, megtekinteni a fiók alapvető adatait, illetve a fiókja jelszavának megváltoztatását a „Saját fiókom” menüpontban.

A láblécben a szoftver logója és szlogenje foglal helyet, valamint egy link, amely ismerteti a felhasználókkal a projekt elképzelését és jellemzőit.

# Minőségi követelmények

Egy hálózati szoftver minőségi követelmények széles skáláját foglalja magában. A szoftver funkcionalitása első sorban a távoli Linux szerverek kezelése, ezért fontos, biztosítani tudja a központi szerver és a távoli szerverek közti biztonságos SSH2 kapcsolatot, a szerver bejelentkezési adatai jogkörüktől függetlenül minden felhasználó előtt rejtve kell, hogy maradjanak. Ezen indokból kifolyólag a szerver jelszava nem kerül az adatbázisban egy mezőben sem eltárolásra, hanem egy random generált számkódról elnevezett fájlra mutat minden adatbázis lekérdezés, amely fájlban base64 titkosítással kerül a jelszó titkosításra. A fájl nem tartalmazza az IP címet, sem az SSH2 portot, így egy illetéktelen rendszerhozzáférés vagy adatbázisból való adatvesztés esetén sem olvashatók a távoli szerverek adatai.

Fontos szempont továbbá még a felhasználói fiókok védelme. Ennek érdekében Argon2ID titkosítást használtam, amely nagyon nehezen feltörhető és ellenáll a brute force támadásoknak, esetleges adatszivárgás esetén a titkosítás önmagában visszafejthetetlen. Továbbá, a bejelentkező felület öt próbálkozás után eltiltja a látogatót a bejelentkezéstől és az elfelejtett jelszópótlási kísérletektől 10 percre. Az ideiglenes eltiltást egy HTTP-cookie végzi. A fiókok védelme érdekében a fiók-jelszónak minimum kilenc karakterből kell állnia, legalább egy számot és egy nagybetűt is kell tartalmaznia.

A fiókok tevékenységelőzményei teljes mértékben visszanezézhetőek, különállóan a fiókműveletek is, illetve különállóan a távoli és helyi szerveren végzett módosítások is. A naplóesemények az adatbázisba kerülnek feltöltésre, azt a szoftver felületén módosítani vagy törölni nem lehet.

A webes tűzfal opcionálisan bekapcsolható, megvédi a webszervert proxy és VPN kapcsolatot használó látogatóktól, felhasználóktól. Ezért arra van szükség, hogy az IP címüket rejtő felhasználók hamis IP címmel ne férjenek hozzá a szoftverhez, hiszen a hamis IP címük kerülne ebben az esetben a naplófájlokba.

# Felhasznált könyvtárak

A szoftver megvalósításához több nyílt forráskódú könyvtár, keretrendszer is felhasználásra került.

- A biztonságos PHP-SQL kapcsolathoz a CodeShack oldalán található minimális SQL könyvtárat vettem igénybe. Nyílt forráskódú és tulajdonosa a honlap felhasználási feltételeiben engedélyezte a használatot, módosítást és a terjesztést is.
- A captcha kód létrehozásához a Secureimage, illetve a Tor böngésző azonosításához TorUtils könyvtárat használtam fel, amelyek egyszerűsített BSD licenccel rendelkeznek, így engedélyezi a terjesztést, módosítást és a felhasználást egyaránt. A két könyvtár szerzője Drew Phillips.
- PHPMailer könyvtárat használtam fel az SMTP csatlakozás létrehozására és a levelek küldésére is egyaránt, amely GNU Lesser General Public licenst használ, amely lehetővé teszi a módosítást, felhasználást és a terjesztést is.
- A reszponzív és kontrasztos megjelenés érdekében az MDB 4 frontend keretrendszert használtam fel, amely MIT licenc alatt publikált: szabadon módosítható és terjeszthető.
- A honlapon látható ikonokat a FontAwesome 5 Free biztosítja, amely Creative Commons Attribution 4.0 licenz alatt teszi lehetővé a módosítást és megosztást.
- Az erőforrások grafikus ábrázolásnál megjelenő diagramokat az ApexCharts könyvtár biztosítja, mely szintén MIT licenz alatt szabadon felhasználható.
- A táblák kényelmes kezeléséhez a DataTables 1.13 JavaScript könyvtárat használtam, mely szintén MIT licenz alatt elérhető.
- A bejelentkezési felület háttéréhez MIT licenc alatt felhasználható ParticlesJS könyvtárat használtam fel. Szerzője Vincent Garreau.
- Az országzászlók ábrázolásához a „flag-icons” nyílt forráskódú, MIT licenz alatt értelmezhető könyvtárat használtam.
- A „font-logos” könyvtárat a Linux disztribúciók megjelenítéséhez használtam fel. Ezen könyvtár nyílt forráskódú, publikus, Unlicense licenc alatt módosítható és szabadon terjeszthető. Szellemi szerzője Lukas W.
- A szerverek térképen való ábrázolására a geoPlugin API-ját használtam, amely Attribution-ShareAlike 3.0 Unported licenc alatt engedélyezi az algoritmus használatát.

# Terjesztés

A szakdolgozatomat GNU General Public License v3.0 licenc alatt teszem elérhetővé. Ez a dokumentum lehetővé teszi a felek számára a kereskedelmi és privát felhasználást, a terjesztést és a módosítást is, azonban a szoftver működéséért nem vállal felelősséget. A terjesztő vagy felhasználó fél köteles feltüntetni a szoftver eredeti forrását és szerzőjét.

## Fejlesztési ütemterv

### Futtatási környezet kialakítása

A futtatási környezet egy Ubuntu virtuális szerveren került kialakításra. Amint kézhez kaptam az ATW Internet Kft. által biztosított szervert és megtörtént az operációs rendszer telepítése, telepítésre került az Apache2 webservert és bekonfiguráltam a HTACCESS hozzáférést is. Ezt követően telepítettem a PHP 8.1 komponenseit, illetve az Argon2 kriptográfiai kiterjesztést is. Ezután a MariaDB szerver és PhPMyAdmin felület került telepítésre, később pedig a KVM virtualizációs rendszer és a hozzá tartozó Libvirt segédeszköz szoftvercsomag.

A futtatási környezet kialakításának befejezéseként beállításra került az Apache VirtualHost, így a webservert konfigurációja a végéhez ért az első indítás utáni három órában.

- 2022. december 1.

### Adatbázis megtervezése

A főbb tervezési feladatok elsődleges szempontja volt az adatbázis megtervezése is. Tekintettel arra, hogy a szoftver minden aktivitást naplóz, így az elsődleges cél az „accounts” tábla megtervezése volt, hogy a többi tábla számára hivatkozási alapot kínáljon a naplózáshoz és a jogosultság megállapításához. Ezután következett a távoli szerverek adatainak eltárolásához használt tábla létrehozása, hiszen ez adja meg a szoftver fő funkcionalitását, ezt



követően a távoli szerverekhez tartozó tevékenységnapló rögzítésére használt tábla került elkészítésre. A virtuális szerverek adatainak eltárolására is készült tábla, illetve az ahhoz tartozó biztonsági mentések eltárolására is.

- 2022. december 2-től 2023. január 1-ig

## Szoftverfejlesztés

Először a szoftver frontend részét alakítottam ki, pontosabban a felső navigációs menüt, majd a bejelentkezési felületet, ezután a vezérlőpultot a hozzátartozó oldalmenüvel. A HTML felület lekódolása, és a hozzátartozó egyedi CSS design megírása három napot vett igénybe.

- 2023. január 1-től 2023. január 4-ig

A szakdolgozatom PHP részét elsősorban az SQL csatlakozáshoz használt osztály kialakításával kezdtem, ezt követően a PHP-SSH2 objektumosztály került kialakításra a távoli szerverek csatlakozásához, majd a szoftver biztonságának biztosítása érdekében létrehoztam a WebApplicationFirewall osztályt is. Későbbi ötlet volt a fejlesztés során a virtuális szerverek létrehozása, ezzel záródik az objektumosztályok fejlesztése.

Az objektumosztályok végeztével következik a komponensek meghívása az oldalakra, és az objektumok példányosítása.

- 2023. január 5-től március 2-ig

## Tesztesetek

A szoftver tesztelése, használata során a rendszer többször összeomlott, amikor olyan távoli szerverre próbáltam csatlakozni a kész rendszerrel, amely offline állapotban van, ezért a RemoteController objektumosztályban a parancsok lefuttatása azzal a feltétellel történik, hogy a szerver a felhasználó által megadott SSH2 porton elérhető, így **a hiba azonnal javításra került.**

- 2023. március 3.

A második hiba a jogosultságkiosztásnál jelentkezett. A felhasználók jogainak kiválasztásakor redundancia lépett fel, így egyes oldalakhoz olyan felhasználók is hozzáférhettek, amelyekhez adminisztrátori jogosultságra volt szükség.

A nem kívánt működést egy logikai hiba okozta, amely a mező kiválasztásakor lépett fel egy INNER JOIN lekérdezés során, ahol a WHERE feltételben rossz változó lett megadva. **A hiba javításra került**, a felhasználói csoportok csak a hozzájuk rendelt funkciókhoz férnek hozzá kizárólag.

- 2023. március 4.

A szoftver további tesztelése során más hibát nem tapasztaltam.

Fejlesztési ütemterv összefoglalása	
Fejlesztési környezet kialakítása	2022. december 1.
SQL adatbázis tervezése	2023. január 1-től 2023. január 4-ig
Frontend fejlesztés	2023. január 1-től 2023. január 4-ig
Backend fejlesztés	2023. január 5-től március 2-ig
Tesztelés és hibajavítás	2023. március 3-4.

# Funkcionalitási terv

## Rendszerszereplők

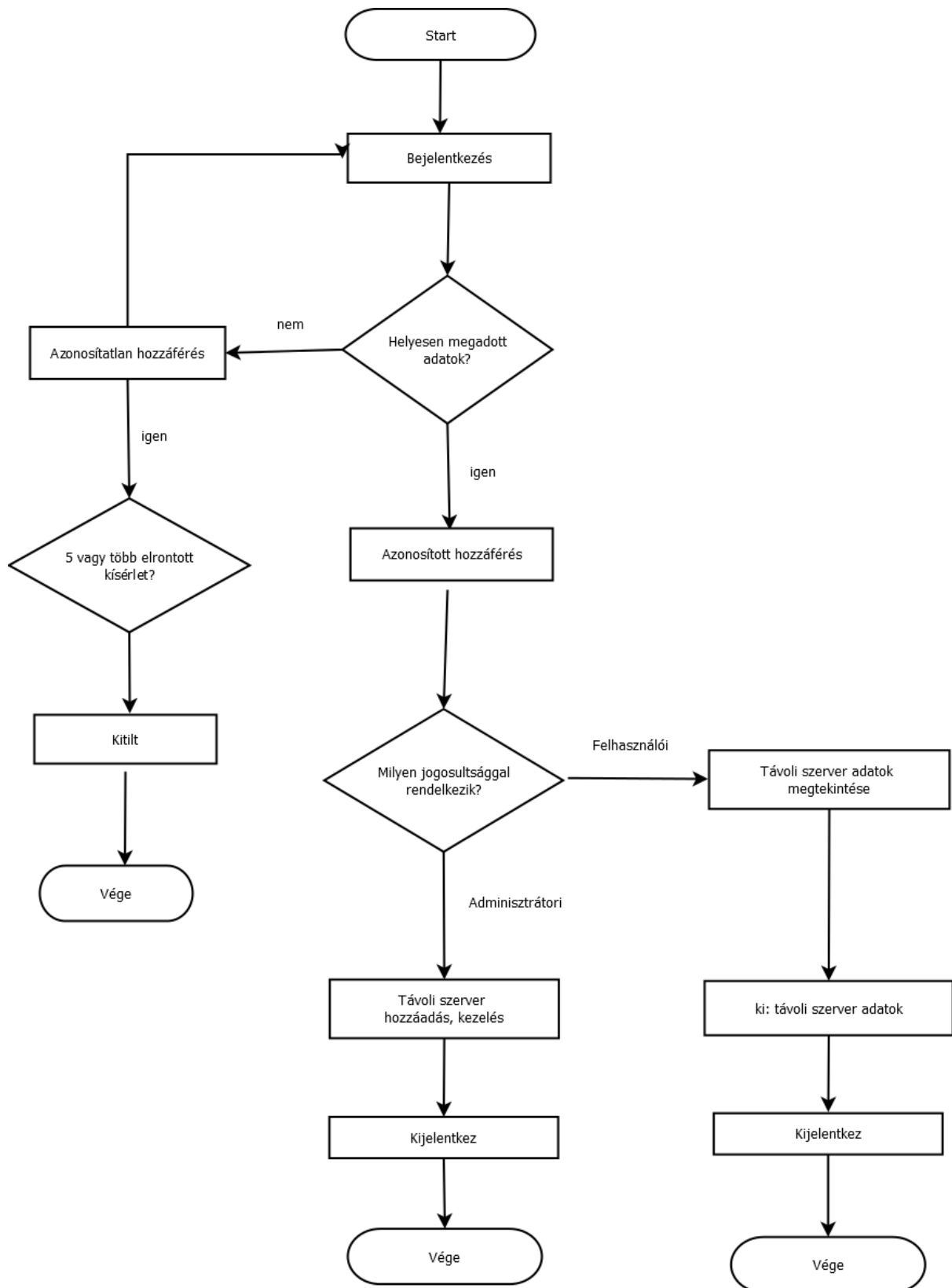
A rendszerszereplők a való életben betöltött szaktudásuk és feladatkörük szerint lettek elkülönítve.

- Adminisztrátor: Tudása vagy képesítése szerint informatikában járatos szakember, főként Linux rendszergazda. Ért az alapvető CLI parancsokhoz és a Linux rendszerfolyamatokhoz.

A szoftverben létrehozó, módosító és törlő műveleteket végezhet: létrehozhat felhasználókat, megtekintheti a tevékenységnaplót, hozzáadhat a rendszerhez távoli szervereket, virtuális szervereket, vagy akár törölheti őket.

- Felhasználó: Az általános felhasználó csak a saját fiókjához fér hozzá, illetve a távoli és virtuális szerverek státuszát képes megtekinteni, továbbá azoknak erőforrás statisztikáit. Bármely olyan munkakör betöltésére létrehozott jogkör, amely nem igényel teljes hozzáférést, csak információgyűjtést.
- Látogató: A szoftver webes felületén böngésző, fiókkal nem rendelkező vagy még be nem jelentkezett, azonosítatlan felhasználó. Szándékairól, szaktudásáról nincs rendelkezésre álló információ, ezért semmilyen művelet végrehajtása nem biztosított a számára.
- Központi szerver: Az a kiszolgáló, amelyre a NextCaligo szoftver és működtető komponensei telepítve vannak. Ezen a kiszolgálón érhető el és használható a felület minden felhasználó számára.
- Távoli szerver: Olyan fizikai számítógép vagy kiszolgáló, amely egy harmadik fél IT infrastruktúrája részeként üzemel.
- Virtuális szerver: Egy olyan virtuálisan létrehozott számítógép, amely rendelkezik saját memóriával, vCPU processzorral és háttértárral. Elindításakor a hozzárendelt ISO lemezképfájllal indul el. A virtuális szerver a központi szerveren foglal helyet fizikailag.

## Folyamatábra<sup>1</sup>



2. ábra: Szoftver folyamatábra

# Implementációs terv

Az implementációs tervben szeretném ismertetni a szakdolgozatom főbb funkciót alkotó algoritmusokat, amelyek nélkül a szoftver nem lett volna megvalósítható jelenlegi formájában.

## Fiókok kezelése

Az elsődleges feladatomban a bejelentkezési rendszer biztosítása volt, az ahhoz tartozó SQL kapcsolat létesítése az Argon2ID titkosítási algoritmus beállítása a jelszavak titkosításához. Több kritériumnak kell megfelelnie a felhasználói fióknak, mint például, hogy a felhasználónév ne legyen foglalt, vagy a jelszó legyen minimum 9 karakter hosszúságú és tartalmazzon legalább egy-egy nagybetűt és számot.

Az Argon2<sup>2</sup> egy 2015-ben kiadott jelszó kriptográfiai módszer, amely jelszófeltörési kísérleteket képes minimalizálni rendkívül erős titkosítási algoritmusával.

```
//Argon2ID titkosítás
$password_hash = password_hash($password, PASSWORD_ARGON2ID,
[
    'memory_cost' => 2048,
    'time_cost' => 4,
    'threads' => 3
]);
```

*3. ábra: Argon2 titkosítás fiók hozzáadása esetén*

Továbbá, hibás bejelentkezési kísérletek esetén a látogatót 5 hibás elrontott email cím és jelszó páros esetén a felület blokkolja a bejelentkezés és a jelszópótlás lehetőségét. Ezt egy PHP SESSION-al viteleztem ki, amelyet egy HTTP-cookie-ba ment el. Ez a „süti” 5 érvénytelen bejelentkezési kísérlet után 10 percre letiltja a bejelentkezést, az idő lejártá után automatikusan törlődik, ezért az algoritmus azt vizsgálja, hogy a megtalálható-e a „süti” „login\_blocked” néven a felhasználó webböngészőjének gyorsítótárában.

Ez a funkció képes az illetéktelen, különféle segédprogramokkal történő bejelentkezések kiszűrésére, ezzel minimalizálva annak veszélyét, hogy egy felhasználói email címmel rendelkező látogató hozzáférjen valaki fiókjához.

```
//Számolja a hibás belépéseket
$_SESSION['logincount']++;

//Hátralévő bejelentkezések kiíratása a látogatónak
$lcremaining = 5 - $_SESSION['logincount'];
$lcremaining_warn = $lang['LcRemaining'];

//Látogató kitiltása 5 elrontott kísérlet után 10 percre
if ($_SESSION['logincount'] > 5)
{
    header('');
    setcookie("login_blocked", "10_minutes", time() + (60 * 10));
    $login_blocked = $lang['LoginBlocked'];
}
}
```

4. ábra: Az elrontott bejelentkezési kísérletek megszámlálása

## Távoli szerverek kezelése

A távoli szerverek elérését biztosító algoritmus a *RemoteController* osztályban foglal helyet, a szerverekre való bejelentkezést és parancsfuttatást a PHP-SSH2 biztosítja. A bejelentkezést az *ssh2\_auth\_password* függvény meghívásával viteleztem ki. A biztonság érdekében a szerverek jelszavát nem adatbázisban, hanem .key formátumú fájlokban tárolom, amelyre az adatbázis fájlnev alapján tud hivatkozni. A jelszófájlok base64 titkosítást használnak.

Az algoritmus megvizsgálja, hogy a megadott IP címen, porton bejelentkezés lehetséges-e, amennyiben igen, tovább lép a felhasználónévvel és jelszóval való bejelentkezéshez, máskülönben pedig a szerveret elérési, hozzáférési, vagy bejelentkezési probléma esetén offline állapotúnak titulálja. Erre azért van szükség, hogy a program ne próbáljon elérhetetlen Linux szerverekre bejelentkezni vagy azon parancsokat futtatni, ugyanis ilyenkor a webszerver az elküldött hálózati kérések száma miatt összeomlana, leállna.

A sikeres parancs elküldése után a kimeneti értéket az algoritmus PHP változóvá konvertálja, amit HTML formátumban megjelenít a felületen, majd lezárja az adatkapcsolatot a szerverrel:

a csatlakozó változó NULL értéket kap. A parancsok lefuttatásához az *ssh2\_exec* függvényt használtam a fejlesztés során.

```
//Csatlakozás és parancs futtatása
public static function ssh_connect($ip_address, $ssh_port, $hash_file, $command)
{

    //key fájl vissza meghívása, tartalmának visszafejtése
    $hash = file_get_contents("/var/www/ssh_keys/{$hash_file}.key");
    $password = base64_decode($hash);

    //SSH kapcsolat létrehozása
    $ssh2_connection = ssh2_connect($ip_address, $ssh_port);

    if (!$ssh2_connection)
    {
        return 0;
    }
    else
    {
        //Bejelentkezés a szerverre
        ssh2_auth_password($ssh2_connection, 'root', $password);

        //Parancs lefuttatása
        $stream = ssh2_exec($ssh2_connection, $command);
        stream_set_blocking($stream, true);
        $stream_out = ssh2_fetch_stream($stream, SSH2_STREAM_STDIO);

        return htmlspecialchars(stream_get_contents($stream_out));
    }
}
```

5. ábra: Kapcsolódás távoli szerverre PHP-SSH2 segítségével

## Virtuális szerverek

A virtuális szerverek működéséért és menedzsmentjéért a KVM osztály felel. Ezen osztály *shell\_exec* függvénnyel futtatja le a rendszerparancsokat. A függvény közvetlen utasítást ad a felhasználó által megadott paraméterekkel rendelkező virtuális szerver kezelésére a Libvirt-nek, amely végrehajtja a HTML felületen megadott műveleteket.

A felületen a kettős műveletvégzés le van tiltva, például a már elindított szervert a felhasználó ne tudja elindítani, hiszen így felesleges parancsokkal terhelné a rendszert.

```
class KVM
{
    //Létrehozás
    //ISO fájlok alapértelmezett helye: /var/lib/libvirt/images
    public static function createVM($name, $vcpu, $memory, $disk_size, $iso)
    {
        $cmd = "/usr/bin/sudo virt-install --connect=qemu:///system --name={$name} --vcpus={$vcpu} --memory={$memory}
        /usr/bin/sudo virsh start {$name} > /dev/null 2>&1";

        return shell_exec($cmd);
    }

    //Státusz lekérdezés
    public static function statusVM($name)
    {
        $cmd = "/usr/bin/sudo virsh domstate {$name}";
        return trim(shell_exec($cmd));
    }

    //Törlés
    public static function removeVM($name)
    {
        $cmd = "/usr/bin/sudo virsh shutdown {$name} > /dev/null 2>&1
        /usr/bin/sudo virsh destroy --domain {$name} > /dev/null 2>&1
        /usr/bin/sudo virsh undefine --domain {$name} > /dev/null 2>&1
        /usr/bin/sudo rm -rf /var/lib/libvirt/images/{$name}.qcow2 > /dev/null 2>&1";
        return shell_exec($cmd);
    }
}
```

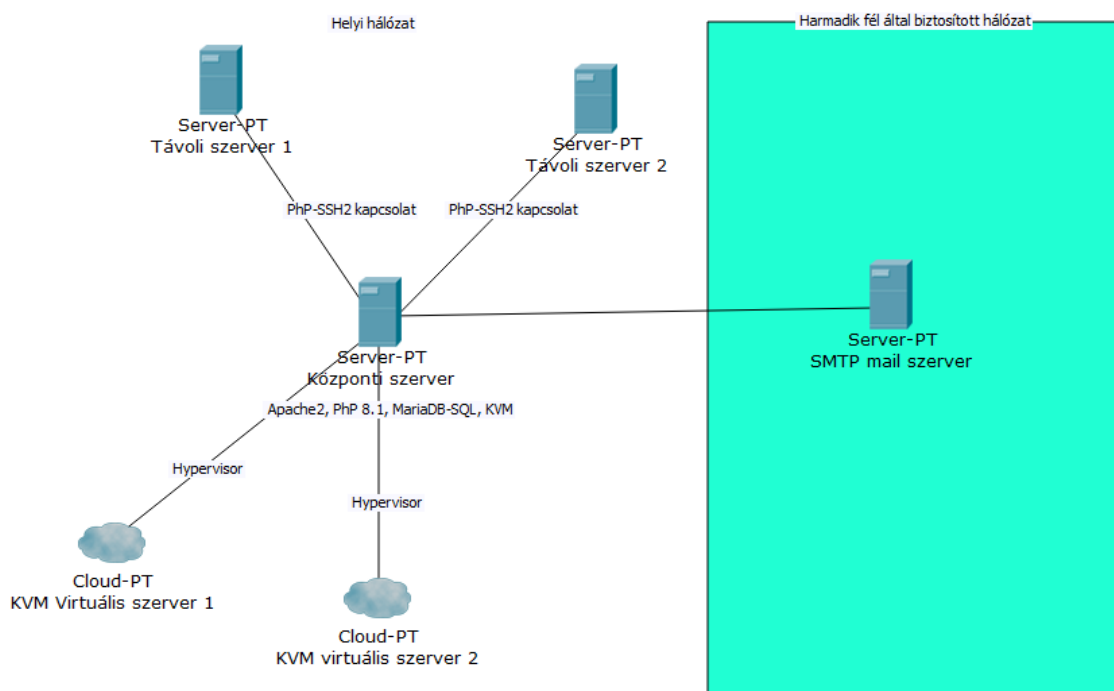
6. ábra: Virtuális gépek kezelése *shell\_exec* paranccsal



# Fizikai környezet

A központi szerver – amelyen a szoftver fő komponensei elhelyezkednek – egy magas rendelkezésre állású Ubuntu bérszerveren (VPS) került elhelyezésre az ATW Internet Kft. szerverparkjában. A bérszerver 2 GB DDR4 memóriával, 1 osztott vCPU processzor maggal és 50 GB SSD tárolóval rendelkezik. Tekintettel arra, hogy a legtöbb fő komponens az SMTP levelező szerver kivételével mind a központi szerveren működnek, ezért a szoftver felépítése szerint monolitikus.

A szoftver teszteléséhez és a szakdolgozat funkcióinak bemutatására két bérszervert vettem igénybe, 1 GB DDR4 memóriával, 1 osztott vCPU processzor maggal és 20 GB SSD tárolóval. A bérszerverek a szoftver funkcióinak demonstrálására készültek, így azokon szoftveres módosítást nem végeztem, az előre feltelepített SSH klienst használtam csak a szoftverhez való csatlakoztatáshoz. A tesztszerverek Kamatera Inc. szerverparkjában kerültek elhelyezésre.



7. ábra: Hálózati topológia

# Felhasználói dokumentáció

## Rendszerkövetelmények

A rendszergazdaként való telepítéshez az alábbi komponensek szükségesek:

- Szerver minimális fizikai hardver követelményei: 2 GB memória, 1 processzormag, 20 GB szabad merevlemez
- Debian alapú – lehetőleg Ubuntu – Linux disztribúció
- Apache2 webservert, HTACCESS hozzáféréssel
- PHP 8.1 csomag és mbstring, mysqli, curl, gd, php-ssh2, libsodium komponensei
- Argon2 titkosítási algoritmus
- KVM és Libvirt szoftvercsomag
- SMTP levelő szerver

Felhasználóként való felhasználásra:

- Bármilyen olyan webböngésző, amely képes a JavaScript futtatására. Ajánlott Mozilla Firefox vagy Google Chrome használata.
- Internethozzáféréssel rendelkező készülék

## A program bemutatása

A NextCaligo szoftver egy webes Linux szerver kezelő felület. A felület lehetőséget nyújt távoli Linux szerverek kezelésére, valamint virtuális szerverek kezelésére, így lehetővé teszi a teljes szerverpark kezelését.

Az adminisztrátori jogkörrel bíró felhasználók képesek a szervereket újraindítani, leállítani, annak jelszavát megváltoztatni, gépnevét módosítani és az erőforrás statisztikákat megtekinteni. Továbbá képesek a naplófájlokat megtekinteni és kezelni a felhasználói fiókokat.

A végfelhasználói jogkörrel rendelkezők a szerverek erőforrás statisztikáinak megtekintésére képesek, illetve megnézhetik a virtuális szerverek státuszát.

A szoftver – jogkörtől függetlenül - mindenkinek lehetőséget kínál saját fiókja hozzáférési adatainak megtekintésére, annak jelszavának módosítására, vagy az elfelejtett jelszó pótlására.

A szoftver használata azon személyeknek ajánlott, akik az IT vagy a távközlés iparában látnak el feladatot, abban járatosak. Főként Linux rendszergazdák, adatközponti technikusok vagy DevOps munkakört ellátó személyek számára lehet kézenfekvő a használata, hiszen egy kis- és középvállalat adatközponti infrastruktúráját lehet vele menedzselni. Alapvető célja, hogy a szervergépeket menedzselni lehessen függetlenül attól, hogy fizikailag hol helyezkednek el a világon: legyenek azok Magyarországon, Európa más területén vagy éppen Amerikában, a felület vezérlőpultjáról egy központi helyről lehetséges irányítani a kiszolgálókat.

# Telepítési útmutató

A felhasználók számára telepítési művelet nem szükséges, a szoftvert a rendszergazdák feladata telepíteni. A telepítés csak Debian Linux alapú disztribúciókon lehetséges manuálisan a rendszerkövetelményeknél felsorolt komponensek telepítésével, majd a szoftverfájlok webszerver főkönyvtárába való mozgatásával, vagy automatizáltan az alábbi terminál parancsokkal:

1. `tar -cvzf nextcaligo.tar.gz`
2. `cd installer`
3. `sudo sh install_hu.sh`

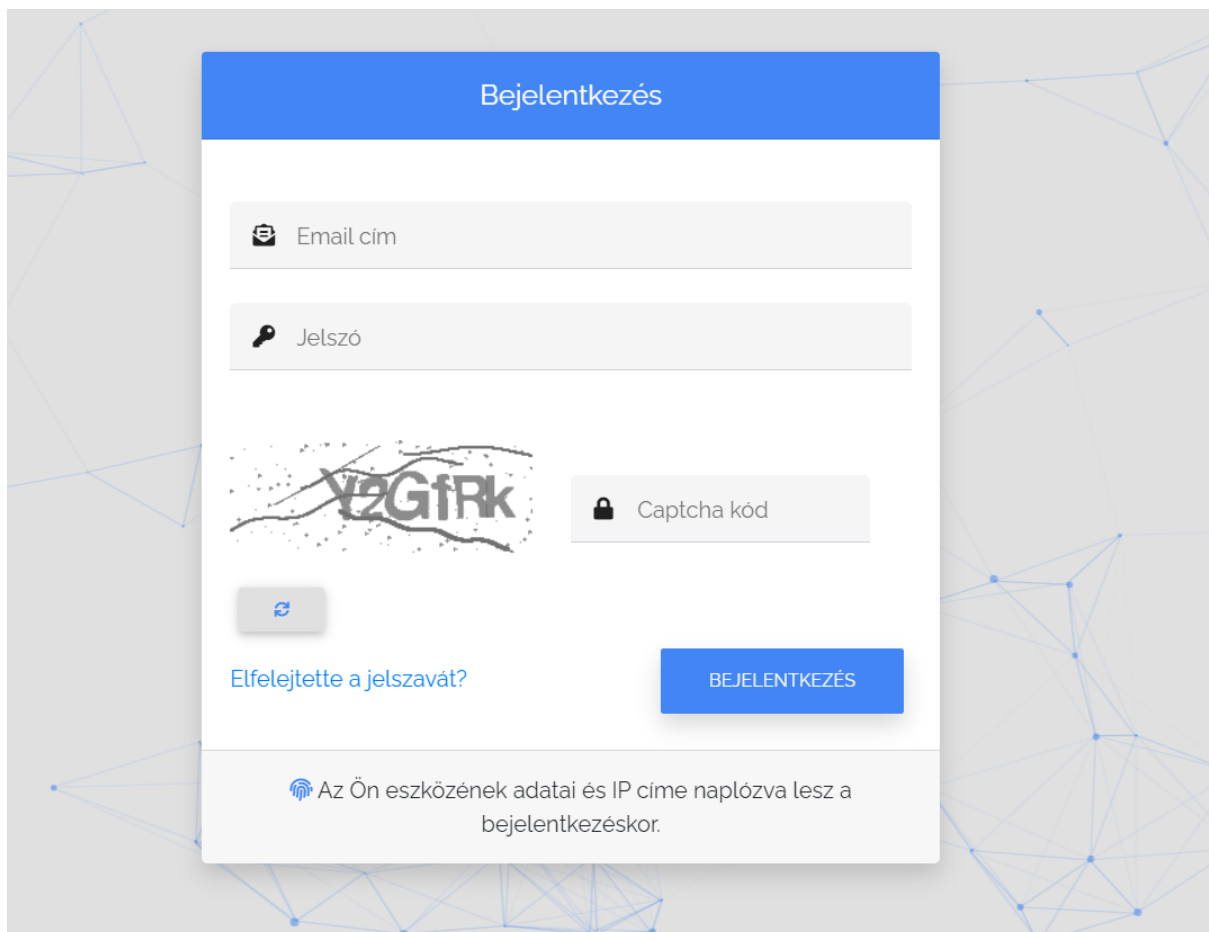
A telepítés végeztével navigáljon a webböngészőjében a szerver IP címén található /install linkre. Itt létrehozhatja felhasználói fiókját, ezt követően törölje a „public” mappából az „install” könyvtárat.

A sikeres telepítést követően bejelentkezhetsz az alkalmazásba és megkezdheti a szerverekkel való munkát.

# Használat

## Bejelentkezés

A program elindítása a bejelentkezési felület megnyitásával történik. A szerverek kezeléséhez és egyéb fiókműveletekhez szükség lesz a személye beazonosítására, hogy a megfelelő jogkörhöz rendelt műveleteket el tudja végezni. Ehhez csak az email címét kell megadnia, illetve a jelszavát, valamint a két mező alatt elhelyezkedő képen látható betűket és számokat is szükséges begépelni a mellette lévő mezőbe. Ezt követően kattintson a „BEJELENTKEZÉS” gombra! A sikeres bejelentkezést követően a rendszer átirányítja a vezérlőpulttra.



The screenshot shows a login form titled "Bejelentkezés" (Login) with a blue header. The form contains the following elements:

- An "Email cím" (Email address) input field with an envelope icon.
- A "Jelszó" (Password) input field with a key icon.
- A CAPTCHA image showing the text "Y2GfRk" with a refresh button (circular arrow icon) to its left.
- A "Captcha kód" (CAPTCHA code) input field with a lock icon.
- A link "Elfelejtette a jelszavát?" (Forgot your password?) in blue text.
- A blue button labeled "BEJELENTKEZÉS" (Login).
- A footer notice: "Az Ön eszközének adatai és IP címe naplózva lesz a bejelentkezéskor." (Your device's data and IP address will be logged at login time.), preceded by a Wi-Fi icon.

8. ábra: Jelszavas bejelentkezés

## Elfelejtett jelszó

Amennyiben elfelejtette jelszavát, annak pótlására is van lehetőség. A bejelentkezési felületen kattintson az „Elfelejtette jelszavát?” linkre, majd az „Email cím” mezőbe írja be a fiókja email címét, illetve a képen látható karaktereket a „Captcha kód” mezőbe, majd kattintson az „EMLÉKEZTETŐ KÜLDÉSE” gombra.


A rendszer egy emailt fog Önnek küldeni, benne egy linkkel, ahol megadhat egy új bejelentkezési jelszót fiókjának. A jelszónak minimum 9 karakteresnek kell lennie, legalább egy-egy nagybetűt és számot kell tartalmaznia. A „JELSZÓ MEGVÁLTOZTATÁSA” gombra kattintva elmenti az új jelszavát és átirányításra kerül a bejelentkezési felületre, ahol már az új jelszóval tud bejelentkezni.

A jelszópótlás után a link érvényét veszti, minden jelszópótlás esetén új jelszót kell kérelmeznie.

## Vezérlőpult

A vezérlőpultban a szoftver funkcióinak összességét láthatja.

## Szerverek menü

A szerverek menüben található táblázatban láthatja a rendszerhez hozzáadott távoli Linux szervereket. A szerverek közt kereshet IP cím, SSH port és régió alapján a táblázat felett elhelyezkedő szöveges mezőben, valamint a  gombra kattintva menedzselheti a szervereket.

## Szerver menedzsment

A felület négy fő menüpontra bontható, amelyek biztosítják a szerver menedzsmentjét.

### 1. Információk

Az információk fül alatt találhatóak a „LEÁLLÍTÁS” és az „ÚJRAINDÍTÁS” gombok.

A kívánt művelet végrehajtásához kattintson a gombra, amely egy felugró ablakban

meg fogja kérdezni, hogy biztosan szeretné-e végrehajtani az adott műveletet. Leállítás esetén a szerver csak fizikailag vagy más felületről indítható újra, ugyanis a szoftver nem fér hozzá a szerverszolgáltató infrastruktúrájához.

A menüpontban látható még a CPU és a memória használat, valamint a szervergép földrajzi elhelyezkedése ábrázolva Google Térképen. A földrajzi elhelyezkedés kiszámítása IP cím alapján történik, ezért pár száz méter eltérés előfordulhat az ábrázolás során.

## 2. Root jelszó

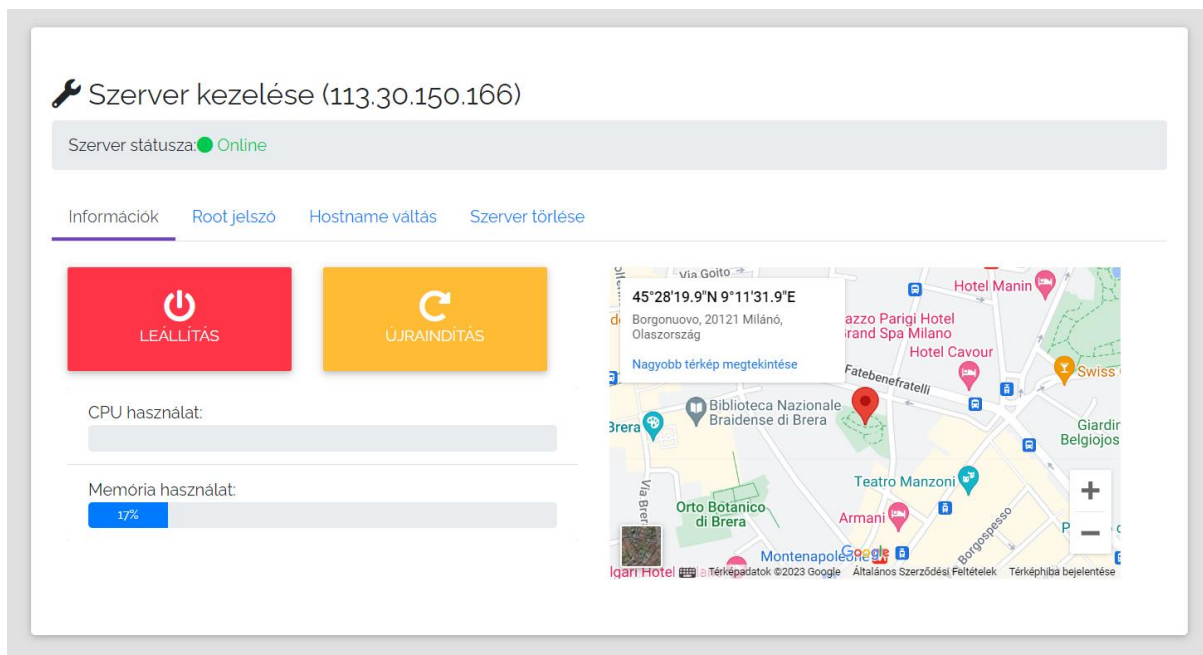
A root jelszó módosításához írja be a felső mezőbe az új jelszót, majd a közvetlen alatta elhelyezkedő mezőbe a jelszó ismétlését. A két jelszónak meg kell egyeznie, minimum 9 karakteresnek kell lennie, valamint tartalmaznia kell legalább egy-egy nagybetűt és számot. Amennyiben meg kívánja változtatni a root jelszót, kattintson a „JELSZÓ MEGVÁLTOZTATÁSA” gombra! A sikeres jelszóváltást követően a megadott jelszóval tud bejelentkezni Linux szerverére.

## 3. Hostname váltás

A hostname a Linux rendszer számítógépneve. Amennyiben módosítani szeretné, az „Új hostname” mezőbe írja be a kiválasztott gépnevet. A mezőben nem használhat különleges karaktereket, valamint az üres karakter használata sem engedélyezett. A módosítás végrehajtása a „HOSTNAME VÁLTÁS” gombbal történik, a sikeres végrehajtás esetén a „Jelenlegi hostname” felirat után megjelenik az Ön által megadott gépnév.

## 3. Szerver törlése

A szerver törlésére lehetőség nyílik a „SZERVER TÖRLÉSE” gombbal. A gombnyomást követően a szerver adatai törlődnek, Önt pedig a szoftver átirányítja a szerverek listájához. A törlés nem visszaállítható művelet.



9. ábra: Képernyőfotó a szerver menedzsmentjéről

## Szerverek hozzáadása

Az oldalmenüben található a „Szerver hozzáadása” menüpont, itt hozzá tudja adni a kezelni kívánt távoli Linux kiszolgálót a rendszerhez. Erre kattintva a „Város vagy régió” mezőbe írja be a város nevét, ahol a távoli szerver földrajzilag elhelyezkedik, majd az alatta lévő legördülő mezőből válassza ki az országot, ahol a szerver fizikailag helyezkedik el. Az „IPv4 cím” mezőbe adja meg a távoli szerver IPv4 címét, valamint az alatta lévő „SSH Port” mezőbe az SSH szerver csatlakozási portját. Ezt követően adja meg a root jelszót, avagy a szerver adminisztrátori jelszavát. A jelszó mező nem tartalmazhat „&” karaktert, mert félrevezetheti a titkosítási algoritmust. Fontos, hogy a hozzáadni kívánt szerver Debian rendszer alapú legyen és rendelkezzen SSH hozzáféréssel, ellenkező esetben a szoftver hibaüzenet megjelenítése mellett abba fogja hagyni a működést.

A szerver hozzáadásához kattintson a „SZERVER HOZZÁADÁSA” gombra! Amennyiben a művelet sikeresen megtörtént, a szoftver ezt egy zöld figyelmeztetőjelzéssel fogja jelezni.



## Szerver hozzáadása

Csak SSH-eléréssel rendelkező Linux kiszolgálók adhatók hozzá a rendszerhez.

Ország




& karakter használata nem engedélyezett a jelszóban.

SZERVER HOZZÁADÁSA

10. ábra: Képernyőfotó a szerver hozzáadásáról

## Virtuális szerverek

A virtuális szerverek listájának megtekintéséhez kattintson a „Virtuális szerverek” menüpontra, ahol a létrehozott virtuális szerverek megtekinthetők egy táblázatban. A virtuális szerverek neve, vCPU magjainak száma, memóriája, merevlemezének mérete és az operációs rendszer logója látható, ezen adatok alapján kulcsszavas keresés a táblázat feletti szöveges mezőben végrehajtható. A virtuális gép menedzsmentjéhez kattintson a  gombra.

## Virtuális szerver létrehozása

Virtuális szerver létrehozásához a fent említett táblázat alatt kattintson a „VIRTUÁLIS SZERVER LÉTREHOZÁSA” gombra, amely át fogja Önt irányítani a megfelelő oldalra.

Írja be a létrehozni kívánt virtuális kiszolgáló nevét a „VM neve” mezőbe. A szöveg nem tartalmazhat üres billentyű leütéseket, különleges karaktereket és már létező virtuális szerver nevét sem. Ezt követően adja meg a vCPU magok számát, a memória méretét MB mértékegységben, majd a merevlemez méretét GB mértékegységben. Ezt követően az alsó legördülő menüből válassza ki az Önnek szimpatikus operációs rendszert – a virtuális gép ezzel a telepítővel fog elindulni az első indításkor, illetve, amíg az operációs rendszer nem került telepítésre. A virtuális szerver létrehozásához kattintson a „Virtuális szerver létrehozása” gombra, a sikeres létrehozást követően át lesz irányítva arra az oldalra, amin a szolgáltatást kezelheti.

## Virtuális szerver kezelése

A virtuális szerver kezelése négy menüpontra osztható, amelyek vízszintesen helyezkednek el a szerver neve alatt található felső menüben.

### 1. Vezérlőpult

A virtuális szervert az „INDÍTÁS” gombbal elindíthatja, a „LEÁLLÍTÁS” gombbal leállíthatja vagy az „ÚJRAINDÍTÁS” gombbal akár is újra is indíthatja. A művelet végrehajtása azonnal meg fog történni, nincs szükség megerősítésre.

### 2. Biztonsági mentés

Biztonsági mentést készíthet a virtuális szerver jelenlegi állapotáról, hogy egy esetleges meghibásodás esetén azt visszaállíthassa. A „Biztonsági mentés neve” szöveges mezőbe írja be a létrehozni kívánt biztonsági mentés nevét, majd kattintson a „BIZTONSÁGI MENTÉS LÉTREHOZÁSA” gombra. Ezt követően a biztonsági mentés meg fog jelenni a gomb alatt található táblázatban.

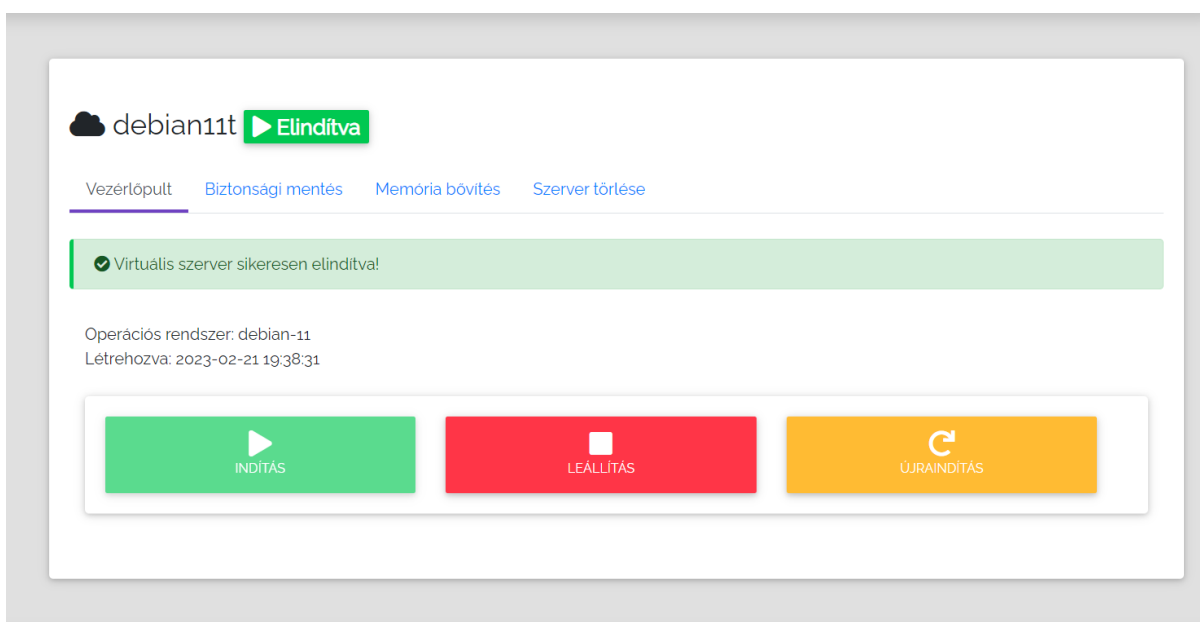
### 3. Memória bővítés

A virtuális szerver memóriájának bővítése lehetséges újraindítás nélkül. Amennyiben szeretné bővíteni a szerver memóriáját, MB mértékegységben írja be az új memória méretét a „Memória (MB)” mezőbe, majd kattintson a „Mentés” gombra. A sikeres

bővítést követően a „Jelenlegi memória” szöveg után meg fog jelenni az újonnan beállított memória mennyisége.

#### 4. Szerver törlése

A szerver törléséhez a képen látható karaktereket gépelje a „Captcha kód” mezőbe, majd kattintson a „SZERVER TÖRLÉSE” gombra. A sikeres törlést követően újra átirányításra kerül a virtuális szerverek listájához. A művelet nem visszaállítható, és a virtuális szerver végleg törölve lesz a kiszolgáló fájlrendszeréből.



11. ábra: Képernyőfotó a virtuális szerver kezeléséről

## Tűzfal & biztonság

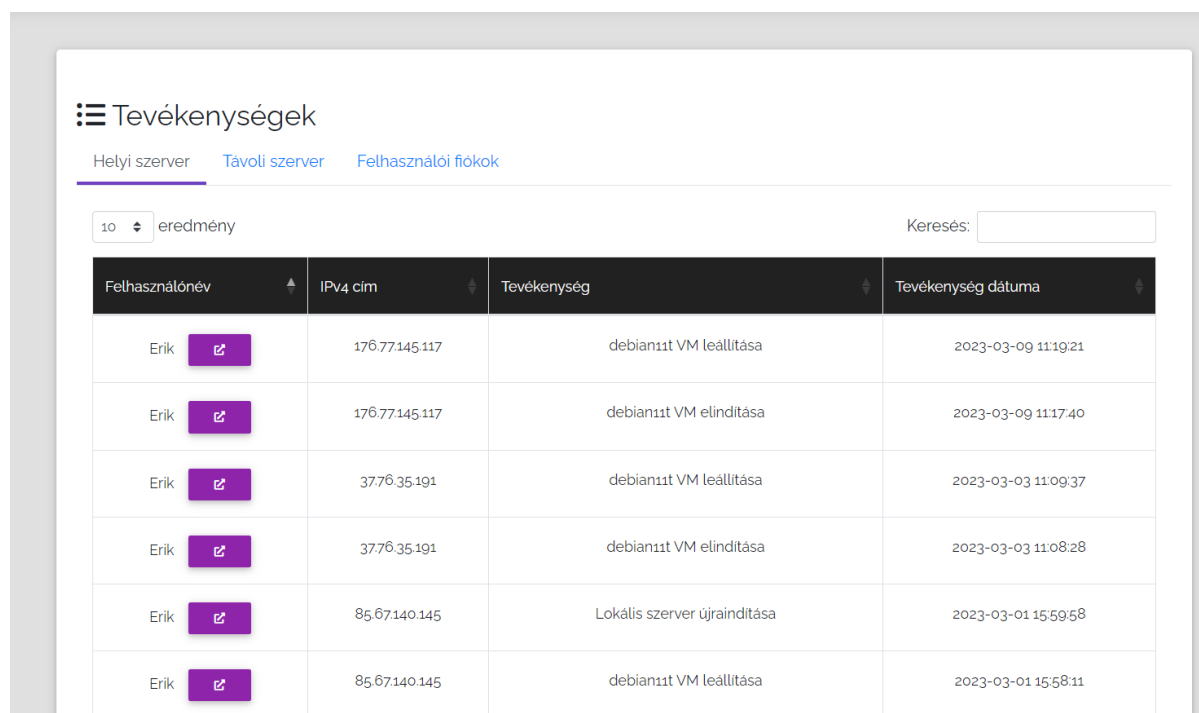
A tűzfal & biztonság menüpontban engedélyezheti vagy kikapcsolhatja a webes tűzfalat. A tűzfal egyes VPN kapcsolatokat képes blokkolni.

A bekapcsoláshoz csúsztassa balról jobbra a bekapcsoló gombot és kattintson a „Mentés” gombra, a kikapcsoláshoz pedig jobbról balra, majd szintén nyomjon a „MENTÉS” gombra.

## Tevékenységek

A tevékenységek menüpontban minden felhasználói tevékenység naplóeseményeit tekintheti meg. A „Helyi szerver” fül alatt látható táblázatban található a helyi, lokális szerveren végzett műveleteket – melyek közé az újraindítás, hostname váltás, root jelszó módosítás, virtuális szerver létrehozás, annak kezelése tartozik, valamint a tűzfal be- és kikapcsolása. A „Távoli szerver”-re kattintva pedig a távoli szerveren végzett módosításokat láthatja. A felhasználók fiókműveleteit a „Felhasználói fiókok” menüpont alatt találja – minden olyan fióktevékenység, amely adatmódosítással járt, az megjelenik ebben a táblázatban.


A táblázat felett elhelyezkedő szöveges mezőben lehetőség nyílik kulcsszavas keresésre, valamint a módosításokat végző felhasználó neve mellett található gombra kattintva megjelenik az adott felhasználó adatlapja, amennyiben Ön adminisztrátori hozzáféréssel rendelkezik.



Felhasználónév	IPv4 cím	Tevékenység	Tevékenység dátuma
Erik	176.77.145.117	debian11t VM leállítása	2023-03-09 11:19:21
Erik	176.77.145.117	debian11t VM elindítása	2023-03-09 11:17:40
Erik	37.76.35.191	debian11t VM leállítása	2023-03-03 11:09:37
Erik	37.76.35.191	debian11t VM elindítása	2023-03-03 11:08:28
Erik	85.67.140.145	Lokális szerver újraindítása	2023-03-01 15:59:58
Erik	85.67.140.145	debian11t VM leállítása	2023-03-01 15:58:11

12. ábra: Képernyőfotó a tevékenységnapló oldalról

## Felhasználók

A „Felhasználók” menüpontban láthatja a rendszerhez hozzáférő összes felhasználót. A felhasználók kezeléséhez kattintson a  gombra.

## Felhasználó kezelése

Az felület bal oldalán láthatja a felhasználó email címét, illetve választhat számára új jogosultsági szintet, majd az „ADATLAP MÓDOSÍTÁSA” gombra kattintva elmentheti az esetleges változtatásokat. A képernyő jobb oldalán található meg a fiókba való legutóbbi sikeres bejelentkezések adatait: IP cím, dátum és készülék típusa – így könnyen azonosíthatja a nemkívánatos hozzáféréseket.

A „Fiók törlése” címszó alatt található „FIÓK TÖRLÉSE” gomb visszaállíthatatlanul, véglegesen törli a fiókot, azzal bejelentkezni, műveletet végrehajtani már nem lehet. A rendszer a naplóbejegyzéseket kivéve minden adatát törli, hiszen a GDPR irányelvei szerint minden felhasználó maga rendelkezik saját adatai felett.

The screenshot displays a user management interface with two main sections. The left section, titled 'Adatlap módosítása (Erik)', contains a form for editing user details. It includes a text input for 'Email cím' with the value 'erikstutz13@gmail.com', a dropdown menu for 'Jogosultság' currently set to 'Felhasználó', and a blue button labeled 'ADATLAP MÓDOSÍTÁSA'. The right section, titled 'Legutóbbi bejelentkezések', features a table with login history. The table has three columns: 'IPv4 cím', 'Dátum', and 'Készülék'. It lists four login attempts, all from Windows devices. At the bottom, there is a red section for 'Fiók törlése' with a warning message and a red button labeled 'FIÓK TÖRLÉSE'.

IPv4 cím	Dátum	Készülék
85.67.140.145	2023-03-09 19:31:36	Windows
37.76.32.64	2023-03-09 09:58:26	Windows
37.76.32.64	2023-03-09 08:53:11	Windows
195.199.113.125	2023-03-06 08:57:20	Windows

13. ábra: Képernyőfotó az adatlap megtekintéséről és a fiók kezeléséről

## Felhasználó hozzáadása

Felhasználói fiókot a „FELHASZNÁLÓ HOZZÁADÁSA” gombra kattintva adhat hozzá a felhasználókat kilistázó táblázat alatt.

Nincs más dolga, mint megadni egy még nem létező felhasználónevet, az „Email cím” mezőbe megadni azt az email címet, amivel azt szeretné, hogy a felhasználó bejelentkezhessen,

majd írja be a bejelentkezéshez szánt jelszót a „Jelszó” mezőbe. A jelszónak legalább 9 karakterből kell állnia, egy-egy nagybetűt és számot kell tartalmaznia. Alul, a legördülő menüből válassza ki a fiók hozzáférési szintjét, miszerint a fióktulajdonos lehet felhasználó vagy adminisztrátor.

Az adminisztrátori fiók rendszergazdai műveleteket képes végrehajtani, a felhasználó pedig csak és kizárólag a szerverek erőforrás statisztikáját képes megtekinteni.

## Központi szerver menedzsmentje

A helyi szerver root jelszavának módosítását tudja végrehajtani ebben a menüpontban. Az „Új root jelszó” mezőbe írja be az új jelszót, majd a „Root jelszó ismét” mezőbe ismét írja be a jelszót. A jelszónak minimum 9 karakteresnek kell lennie, legalább egy-egy nagybetűt és számot kell tartalmaznia.

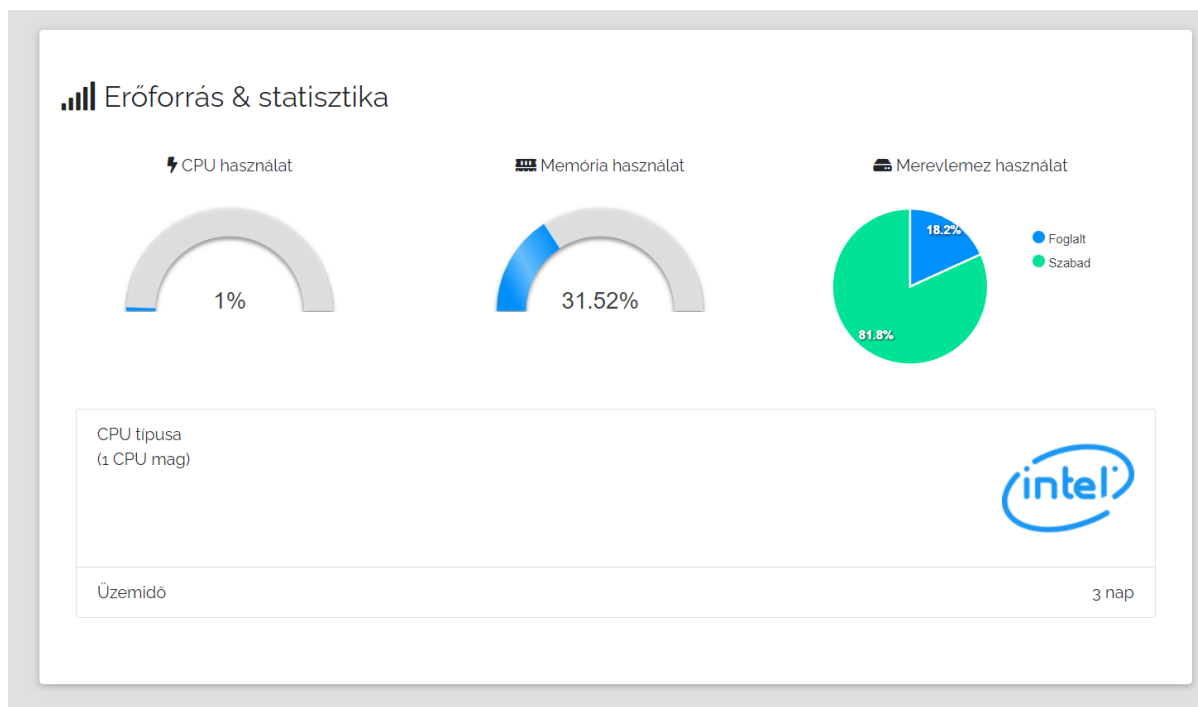
A jelszó megváltoztatásához kattintson a „JELSZÓ MEGVÁLTOZTATÁSA” gombra. A sikeres jelszóváltást követően ezzel a jelszóval tud bejelentkezni root felhasználóként a Linux rendszerbe.

A hostname avagy számítógépnév megváltoztatásához írja be az „Új hostname” mezőbe a beállítani kívánt gépnevet, majd kattintson a „HOSTNAME VÁLTÁS” gombra. Az új gépnév nem tartalmazhat speciális karaktereket.

Az újraindítás gombra kattintva tudja a lokális, központi szervert újraindítani. A művelet végrehajtásához kattintson az „ÚJRAINDÍTÁS” gombra. A szoftver meg fogja kérdezni egy felugró ablakban, hogy biztosan végre akarja-e hajtani a műveletet. Itt ismét kattintson az „ÚJRAINDÍTÁS” gombra.

## Erőforrás & statisztika

Ezen menüpontban megtekinthető a CPU, memória használat %-ban, illetve a merevlemez kihasználtságát, a lefoglalt és a szabad tárhelyét. Továbbá a CPU magok számát, a processzor márkáját és a Linux rendszer üzemidejét.



14. ábra: Képernyőfotó az erőforrás statisztikáról

## Saját fiókom

A „Saját fiókom” menüpontban megtalálhatja a saját fiókja adatlapját, annak alapértelmezett adataival: felhasználónév, email cím, fiók létrehozásának dátuma.

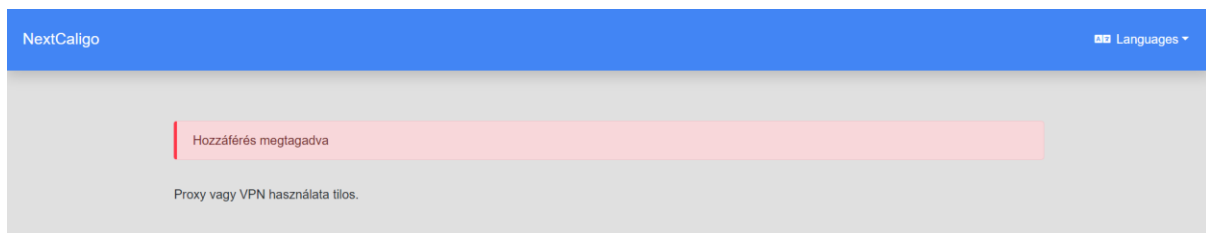
A jelszó megváltoztatására kattintva megjelenik a fiók jelszavának módosításának lehetősége. A „Jelszó” mezőbe írja be a fiókjához beállítani kívánt jelszót, majd kattintson a „JELSZÓ MEGVÁLTOZTATÁSA” gombra. A sikeres jelszóváltást követően a megadott jelszóval tud bejelentkezni a fiókjába.

A fiókból való kijelentkezéshez a felső menüben kattintson a „Kijelentkezés” linkre, majd átirányításra kerül a bejelentkezési felületre.

# Hibakezelés

Előfordulhat, hogy a weboldalhoz való csatlakozáskor a szoftver a „Hozzáférés megtagadva” hibaüzenetet közvetítheti. Ennek oka az lehet, hogy Ön Tor böngészőt, vagy egyéb más, olyan böngészőt használ, amelyben – akár előre – telepített VPN vagy Proxy kapcsolatot biztosító bővítmény van.

Ennek megoldása a tiltott kapcsolatokat biztosító bővítmény törlése, majd a böngésző újraindítása, vagy másik böngésző használata.



*15. ábra: Képernyőfotó a "Hozzáférés megtagadva" hibaüzenetről*



# Összegzés

Úgy gondolom, a szakdolgozat előzetes tervezetéhez képest minden -, a szoftverrel és magammal szemben állított – elvárást teljesítettem.

A fejlesztésről volt egy előzetes tervezetem, hiszen a szerverek és a felhő témakörével évek óta foglalkozom. Sikerült minden eltervezett funkciót a szoftverbe implementálni, és személy szerint úgy érzem, a fejlesztés közben felbukkanó hibák hozzájárultak nem csak a szoftver, de az én szakmai fejlődésemhez is. Sőt, a virtuális szerverek létrehozására és kezelésére alkalmas funkciót fejlesztés végén építettem a szoftverbe, mert a működő komponensek motiváltak arra, hogy tovább fejlesszem az alkalmazást. Ettől függetlenül, rengeteg továbbfejlesztési lehetőséget látok az alkalmazásban, mind a hálózati biztonság, mind pedig a felhasználói kényelem terén – ezt alább, a fejlesztési lehetőségeknél fejtem ki.

A legnagyobb nehézséget az SQL adattáblák összekapcsolása okozta, ugyanis volt, hogy három adattáblát is össze kellett kapcsolnom egyetlen lekérdezésben. A projekt végéhez érve sikerült ezt a logikát elsajátítanom és ezeket a lekérdezéseket használni.

# Fejlesztési lehetőségek

Kifejezetten sok lehetőséget látok a szoftver továbbfejlesztésében: a felhő alapú, több szerveres tematikájú szoftverekre igény van a piacon a technológia fejlődésével. Éppen ezért nagyobb hangsúlyt fektetve az adatbiztonságra és a virtualizációra egy VPS menedzsment platform létrehozásában, fejlesztésében gondolkodom. Különös tekintettel arra, hogy Magyarországon található IT szolgáltatók jelentős része bérelt szoftvereket használ az infrastruktúrája kezelésére, sokszor ezért megszólalásig hasonló felületekkel találkozhat az, aki szeretne bármilyen szerver hoszting szolgáltatást igénybe venni. Úgy gondolom, hogy egy új IaaS (Infrastructure as a Service) platformra igény van a piacon az IT vállalatok számára.

A szoftverem további használatát hoszting alkalmazásként tudom elképzelni, és ennek megfelelően továbbfejleszteni. Elérhetővé tenném a végfelhasználói fiók szabad regisztrációját, majd hozzáadnék további biztonsági funkciókat: két lépésős azonosítás, IP cím alapú vagy Python integrációval arckép alapú bejelentkezés.

A következő elképzelésem a virtualizáció és a szerver menedzsment terén hozna új lehetőségeket. VMware alapú virtuális szervereket hozhatnának létre a regisztrált felhasználók. A létrehozás során a VPS operációs rendszerét, illetve a földrajzi elhelyezését is ki lehetne választani, majd a szolgáltatás létrejöttével a felhasználók PayPal integráció segítségével tudnák a szolgáltatás díját automatizáltan kifizetni.

# Köszönetnyilvánítás

Végzőként, ezúton szeretnék köszönetet mondani témavezetőmnek, Horváth Attila tanár úrnak a szakdolgozat megírásához szükséges mentorálásáért, illetve szeretném köszönetem kifejezni minden eddigi pedagógusnak és osztálytársamnak, aki segített az évek során a szoftver kifejlesztéséhez szükséges logikát, szakmai tudást építő kritikáival és ötleteivel elsajátítani.

# Források

[1]: A szoftver folyamatábráját az ingyenes Dia 0.97.2 programmal terveztem:

<http://live.gnome.org/Dia>

[2]: Wikipédia: <https://en.wikipedia.org/wiki/Argon2>

# Ábrajegyzék

1. ábra: SQL adatbázis táblák és logikai kapcsolataik .....	12
2. ábra: Szoftver folyamatábra .....	20
3. ábra: Argon2 titkosítás fiók hozzáadása esetén .....	21
4. ábra: Az elrontott bejelentkezési kísérletek megszámlálása .....	22
5. ábra: Kapcsolódás távoli szerverre PHP-SSH2 segítségével.....	23
6. ábra: Virtuális gépek kezelése shell_exec paranccsal.....	24
7. ábra: Hálózati topológia .....	25
8. ábra: Jelszavas bejelentkezés .....	29
9. ábra: Képernyőfotó a szerver menedzsmentjéről.....	32
10. ábra: Képernyőfotó a szerver hozzáadásáról .....	33
11. ábra: Képernyőfotó a virtuális szerver kezeléséről.....	35
12. ábra: Képernyőfotó a tevékenységnapló oldalról .....	36
13. ábra: Képernyőfotó az adatlap megtekintéséről és a fiók kezeléséről .....	37
14. ábra: Képernyőfotó az erőforrás statisztikáról .....	39
15. ábra: Képernyőfotó a "Hozzáférés megtagadva" hibaüzenetről .....	40