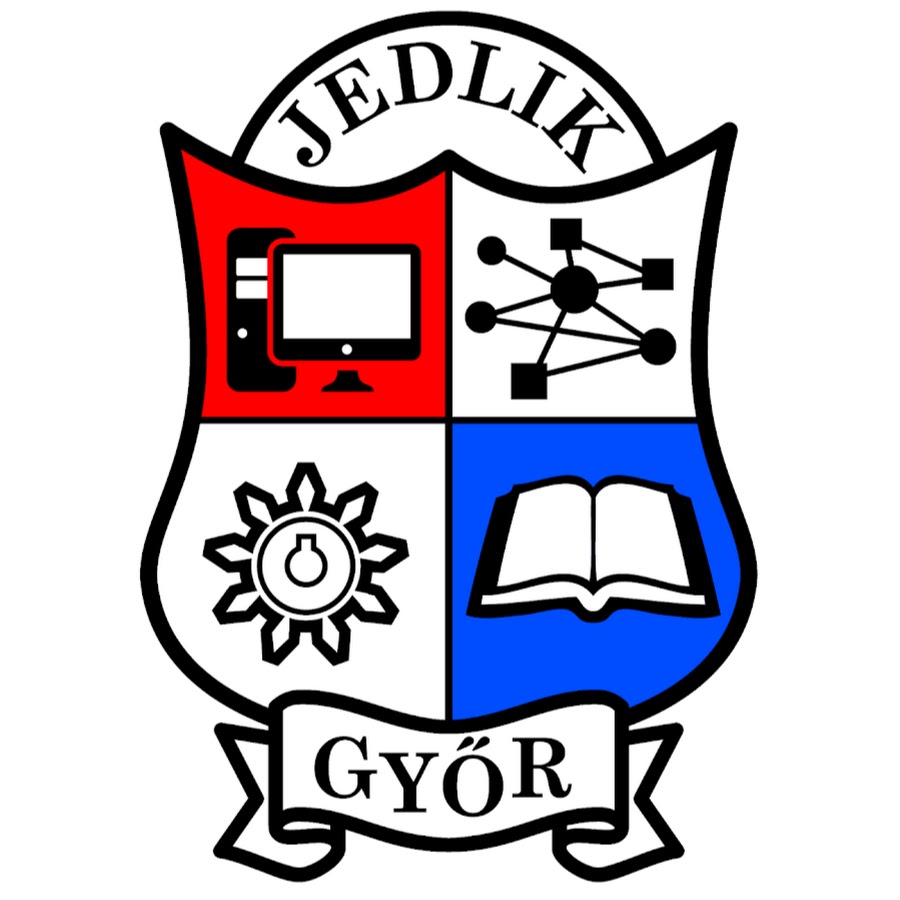
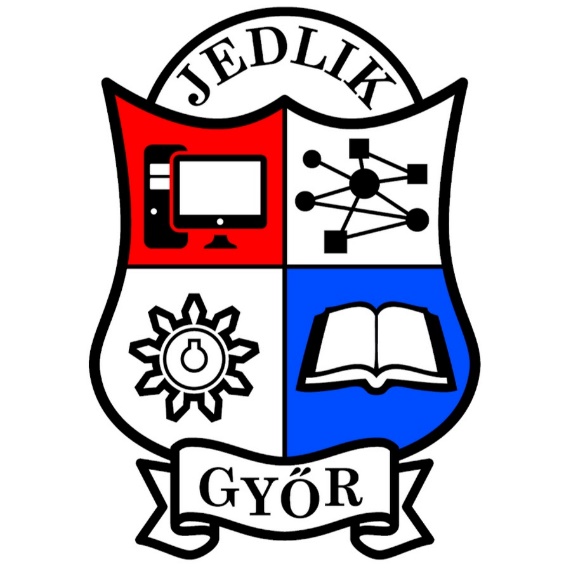
Kovács Bence és Szabó Tamás

Szakdolgozat





# Full-stack webshop

**(Node.js, Vue.js, Firebase)**

**Készítette**

Szabó Tamás Martin Kovács Bence Dominik

*Győr, 2022*

Tartalomjegyzék

[Fedőlap 1](#_Toc101890599)

[Full-stack webshop 2](#_Toc101890600)

[Bevezető 3](#_Toc101890601)

[Szakdolgozatunk tartalma egy letisztult, egyszerűen kivitelezett webshop. Aminek a szerveroldali kiszolgálását (Backend) Node.Js szoftverrendszer látja el. A felhasználóval közvetlen kapcsolatba levő (Frontend) megvalósításához a Vue.JS keretrendszer segítette a munkánkat. Az adatbázis létrehozását, kialakítását és kivitelezését pedig a MongoDB dokumentumorientált adatbázis szoftver tette lehetővé. 3](#_Toc101890602)

[Azért választottuk ezeket a szoftvereket és keretrendszereket, mivel tanulmányaink során nagyon szimpatikussá vált a használatuk és tanulási nehezségük, illetve könnyen találtunk hozzá segédanyagokat és oktató videókat az interneten. 3](#_Toc101890603)

[Alapvetően a munkamegosztás nem okozott nehézséget számunkra, mivel a tervezési fázisban tudtuk, hogy ki melyik részével szeretne foglalkozni a projektnek, viszont, ha elakadtunk akkor a másik segítségével megoldottuk az adott problémát. A munka gördülékenyen ment. 3](#_Toc101890604)

# Bevezető

# Napjainkban nagyon sok ember szeret vásárolni akár üzletekben, akár nagyobb üzletláncokban, kisebb kereskedésekben, de ez alapvetően mindenki számára egy fárasztó tevékenységnek minősül, mivel el kell menni otthonról megvenni, amit szeretnénk és ez sok időt fel tud emészteni. Ezért napjaink szerves részévé váltak az internetes boltok és üzletek (webshopok) amik megkönnyítik az emberek számára a vásárlást, és nem beszélve sok időt tudunk vele megspórolni. Az internetezők négyötöde online vásárló, 43%-uk legalább havonta vásárol, évente átlagosan 12 alkalommal.

# Szakdolgozatunk tartalma egy letisztult, egyszerűen kivitelezett elektronikai üzletnek webshopja. Aminek a szerveroldali kiszolgálását (Backend) Node.Js szoftverrendszer látja el. A felhasználóval közvetlen kapcsolatba levő (Frontend) megvalósításához a Vue.JS keretrendszer segítette a munkánkat. Az adatbázis létrehozását, kialakítását és kivitelezését pedig a MongoDB dokumentumorientált adatbázis szoftver tette lehetővé.

# Azért választottuk ezeket a szoftvereket és keretrendszereket, mivel tanulmányaink során nagyon szimpatikussá vált a használatuk és tanulási nehezségük, illetve könnyen találtunk hozzá segédanyagokat és oktató videókat az interneten. Alapvetően a munkamegosztás nem okozott nehézséget számunkra, mivel a tervezési fázisban tudtuk, hogy ki melyik részével szeretne foglalkozni a projektnek, viszont, ha elakadtunk akkor a másik segítségével megoldottuk az adott problémát. A munka gördülékenyen ment.

# ****Szoftver alapvető jellemzői****

# ****Project neve:** Electrical shop**

# ****Project kategóriája:** Webshop**

# ****Project főbb funkciói:** 1. Oldalhoz tartozó felhasználó létrehozása, tárolása, validálása**

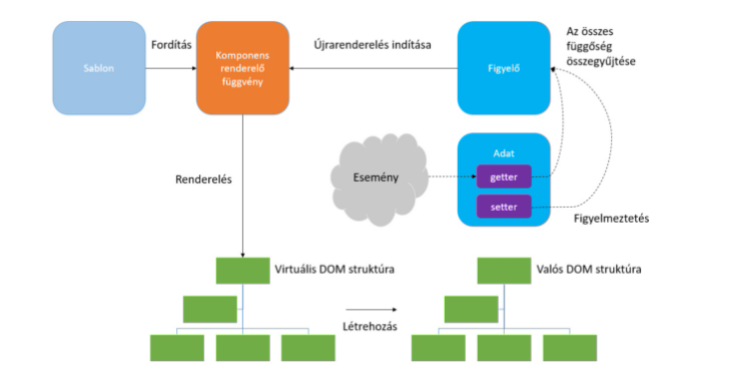
# **2. Termékek felvétele, tárolása, szerkeszthetősége, állapot frissítése.**

# **3. Kosárban található termékek összegzése, rendezése**

**Használt szoftverek, keretrendszerek**

**Vue.js**

A Vue.js egy reaktív JavaScript keretrendszer. A legfontosabb funkciója az adat összekötés **(data binding).** Segítségével létrehozhatunk egy oldalas weboldalakat, valamint komponenseket is hozhatunk létre. A Vue.js az fejlesztői szerint egy nélkülözhetetlen eszköz a modern webfelületek hatékony fejlesztésében.

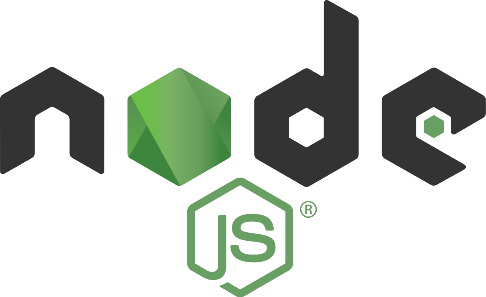
**A Vue.js egy HTML-alapú sablon szintaxist használ, amiben lehetőségünk adódik, hogy összekössük a lerenderelt DOM-t a lepéldányosított Vue-val kapcsolatos adatokat.**

Számos JavaScript keretrendszer, beleértve a Vue.js-t is az úgynevezett Virtuális DOM-t használja. Elméletileg a DOM-ra gondolhatunk úgy, ami reprezentálja a HTML dokumentum felépítését. Gyakorlatilag a DOM egy olyan adatstruktúra ami akkor jön létre, amint a HTML dokumentumot a böngésző lefordítja. **virtuális DOM renderelő függvényekké.**



**Node.js**

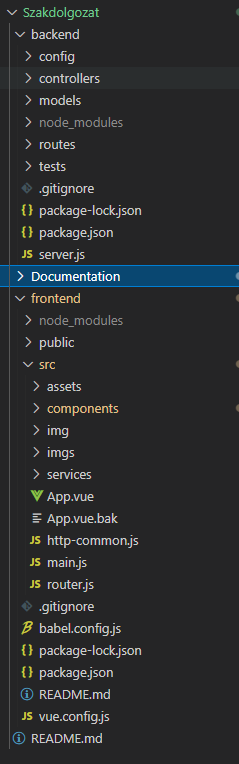
Node.Js egy olyan futtató környezet, ami lehetőséget nyújt JavaScriptben írt programok futtatására szervereken. JavaScriptet leginkább a kliens oldalon szokták használni, a böngészőn keresztül. A böngésző applikáció fejlesztők számára olyan motorokat kell biztosítani, amik a JavaScript kódot gépi nyelvként interpretálják és le is futtatják.

Annyi könnyebséget nyújt, ha már dolgoztunk frontend oldalon, nem kell egy teljesen új nyelvet megtanulni ahhoz, hogy a backend-et is implementáljuk. A másik előnye az, hogy a non-blocking természetéből adódóan alkalmas egyszerre több ezer bejövő 'egyszerű' kérés kezelésére.

**MongoDB**

**A MongoDB nyílt forráskódú dokumentumorientált adatbázis szoftver, amelyet a MongoDB inc. fejleszt. A NoSQL adatbázisszerverek közé tartozik. A dokumentumokat JSON-szerű formátumban tárolja (BSON). A MongoDB-t olyan nagyobb felhasználók is használják.**

# Frontend (Vue Js)

**Komponensek, mappa struktúra**A fő szempont, amit követtünk az, hogy külön komponensekre tudjuk bontani az oldalt és egy olyan mappa struktúrát tudjunk létrehozni, ami átlátható és rendszerezett és könnyen lehessen vele dolgozni.

A projektet először két fő szekcióra bontottuk, ami nem lett más, mint a backend és a frontend mappa. Ezáltal elértük, hogy a munkánk teljesen átlátható legyen. Ezeken belül megtalálhatóak a keretrendszerhez tartozó mappák.