Miskolci Egyetem

Gépészmérnöki és Informatikai Kar

Általános Informatikai Intézeti Tanszék



**Ételajánló webes alkalmazás**

**Szakdolgozat**

**Készítette:**

**Név:** Szilágyi Gábor

**Neptunkód:** S9NJK6

**Szak:** Mérnök Informatikus BSc

Korszerű web technológiák szakirány

**1. Bevezetés**

A mindennapjaink egyik fontos része az evés és az ezt megelőző döntéshozatal. A mai választási lehetőségekben, egyre több emberrel esik meg, hogy nem tudja eldönteni mit egyen. Ezen a helyzeten az sem segít, hogy az embereknek egyre kevesebb idejük van arra, hogy saját maguknak főzzenek.

Szakdolgozatom célja olyan webalkalmazás elkészítése, ami ezt a mindennapi döntéshozatalt és a velejáró extra munkát könnyíti meg. Az alkalmazás a felhasználó által megadott város és konyha alapján étel és étteremajánlást biztosít kérdések segítségével. A felhasználó ugyanakkor segíthet az opciók bővítésében azzal, hogy ezeket elküldi az erre megadott felületen az adminisztráció felé. Az adminisztrációnak ugyanakkor biztosított egy olyan felület ahol kezelheti az adatbázis adatait.

A szoftver megvalósítására a Spring és az Angular keretrendszereket választottam. A Spring keretrendszer Java programozási nyelvű alapokra épül, az Angular pedig a JavaScript nyelvre. Ezen keretrendszerek a szoftver fejlesztését nagyban megkönnyítik a különböző, már előre implementált szolgáltatásaikkal. Választásomat nagyban befolyásolta az a tény, hogy ezek a keretrendszerek, technológiák az iparban is keresettek. Adatbázisnak a MongoDB-t választottam, mivel az alkalmazáshoz nincs szükség egy relációs adatbázisra és a funkciók implementálását nagyban megsegíti egy már létező és jól személyre szabható keretrendszer használata.

Az alkalmazás használatához a felhasználóknak és az adminisztrátoroknak nem szükséges komolyabb informatikai tudással rendelkezniük. A felhasználóknak nem szükséges sok inputot megadniuk, mert a választási lehetőségek előre adott igen/nem válaszok, valamint a települések és konyhák neveinek listája is adott. Az adminisztrációs felület annyiban bonyolultabb, hogy az adminisztrátorok az adatbázist kezelik, ezért az inputok nagy része szöveges. A szöveges formátumok miatt lehetnek olyan kérések, amelyek hibás formátumúak.

**2. Felhasznált eszközök, technológiák**

**2.1. Felhasznált eszközök**

**2.1.1. InteliJ Idea**

Az *InteliJ* Idea a JetBrains által fejlesztett Java alapú integrált fejlesztői környezet. Arra fejlesztették ki, hogy nagyban megkönnyítse a fejlesztők feledatát, akik Javában, Kotlinban, Groovyban és más JVM alapú nyelvben fejlesztenek. Nagyon népszerű ez a fejlesztői környezet, hiszen sok segítséget nyújt a programozónak és a számtalan könnyen elérhető biztonságos plugginjaival nagy testreszabhatóságot biztosít.

A választásom azért erre a programra esett, mivel az iparban is nagyon sok helyen használják, könnyű magának a programnak a használata és én magam is már évek óta használom, így jobban ismerem, mint a többi versenytársát.

**2.1.2. WebStorm**

A *WebStorm* szintén egy JetBrains által fejlesztett IDE. A *WebStorm* JavaScript és a hozzá kapcsolódó nyelvekkel való programozás megkönnyítésére lett kifejlesztve. Hasonló az *InteliJ*-hez mind a felhasználó felületében és kezelhetőségében valamint abban, hogy ugyan úgy biztosítja számunkra a nagy testreszabhatóságot.

Azért esett erre a programra a választásom, mert nagyban hasonlít a már említett *InteliJ*-hez és szintén már évek óta használom ezzel megkönnyítve a programom fejlesztését.

**2.2. Felhasznált technológiák**

**2.2.1. MongoDb**

A MongoDB egy nyílt forráskódú NoSQL adatbázis kezelő rendszer, ami egy jó alternatíva a relációs adatbázisokra. Tipikusan nagymennyiségű adatok tárolására és ezek kezelésére szokták használni és dokumentum orientált adattárolásra és kezelésre is képes. Szervezetek még többek közt ad-hoc lekérdezésekre, terhelésmegosztásra és szerver oldali JavaScript futtatásokra is szokták használni.

NoSQL mivoltából a MongoDB nem táblázatszerű architektúrát használ, mint egy relációs adatbázis, hanem gyűjteményekből és dokumentumokból architektúrát. A dokumentumok kulcs-érték párokat tartalmaznak, míg a gyűjtemények (SQL tábláknak felelnek meg) dokumentumok kollekcióit.

A dokumentumok hasonlítanak a JSON-hoz de a MongoDB ennek egy variációját használja, ami a BinaryJSON (BSON). Ennek az előnye az, hogy sokkal több féle adattípust támogat ezzel megkönnyítve a fejlesztők munkáját. A tárolt adatok lehetnek akár másik dokumentumok, tömbök, de akár ezeknek a kombinációja is a felhasználói kézikönyv szerint.

Tehát a MongoDb összegezve:

Miért használják?

* Nagymennyiségű strukturált és strukturáltalan adat tárolása függőleges és vízszintes skálázhatósággal. Keresés mezővel, indexel és kifejezéses lekérdezésekkel.
* Integrált adatok alkalmazásokhoz, beleértve hibrid és többfelhős alkalmazásokat.
* Az adatbázis lehetővé teszi a beágyazott struktúrákat.
* Több szerveren lehet futtatni egyszerre.

Előnyei:

* Séma nélküli, ami lehetővé teszi akármilyen típusú adat tárolását lehetővé téve ezzel az egyszerű skálázhatóságot és nagyobb flexibilitást relációs adatbázisokhoz képest.
* Dokumentum orientált, ami egyszerűbbé teszi a más nyelvekben objektumokká való átfordítást és csökkenti az adatbázis join-ok használatát ezzel csökkentve a költségeket.
* Egyszerű skálázhatóság támogatva az adatbázis szilánkosítását és az adatok zónákba való csoportosítását kulcsokkal.
* Harmadik féltől származó tároló motorok támogatása.
* DBMS-be közvetlen beépített aggregációs lehetőség.

Hátrányai:

* Az automatikus feladatátvételi stratégiájának köszönhetően, ha a mester node nem működik, akkor egy másik node átveszi a feladatát ezzel folytonosságot biztosítva, de ez a folyamat akár egy percet is eltarthat.
* Egy mester node használata miatt az írási mennyiség is limitálva van ezzel potenciális torlódást okozva.
* Nem biztosít, teljes referenciális integritást mivel nem használ idegen kulcsokat.
* Felhasználó hitelesítés nincs alapértelmezetten engedélyezve ezzel rést hagyva potenciális hackereknek bár hozzá lett adva, hogy egy konfigurálatlan adatbázis letilt minden hálózati kapcsolatot.

Továbbá a MongoDB sok programozási nyelvet is támogat, többek között a C, C++, C#, Java, Ruby és a Swift.