

VILNIAUS GEDIMINO
TECHNIKOS UNIVERSITETAS
ELEKTRONIKOS FAKULTETAS
ELEKTRONINIŲ SISTEMŲ KATEDRA

Atvirojo kodo įranga mokslui, verslui ir valdymui

Namų darbas

Atliko: Domantas Skuja AKSfm-20

Tikrino: Raimond Laptik

Vilnius, 2021

Namų darbo užduotis

Surasti atviro kodo web serverį ir atviro kodo duomenų bazę. Įdiekite, sukonfigūruokite ir paleiskite web serverį ir duomenų bazę jūsų atviro kodo operacinėje sistemoje. Sukurkite puslapį, kuris galėtų nuskaityti tekstą iš duomenų bazės. Pridėkite į duomenų bazę bet kokį tekstą ir įsitikinkite, kad jis atvaizduojamas. Užpildykite namų darbo ataskaitą, išsamiai aprašant visus žingsnius. Argumentuokite programinės įrangos parinkimą.

Naudojama programinė įranga

Apache

Apache HTTP Server yra nemokama atvirojo kodo daugiaplatformė serverio programinė įranga, išleista pagal „**Apache License 2.0**“ licenziją. „Apache“ kuria ir prižiūri atvira kūrėjų bendruomenė, globojama „Apache Software Foundation“. Didžioji dauguma Apache HTTP Server egzempliorių veikia Linux operacinėje sistemoje, tačiau dabartinės versijos taip pat veikia Microsoft Windows, OpenVMS ir įvairiose į Unix panašiose sistemose. Ankstesnės versijos taip pat veikė NetWare, OS/2 ir kitose operacinėse sistemose, įskaitant prievadus į pagrindinius kompiuterius.

MariaDb

MariaDB Server yra viena iš populiariausių atvirojo kodo reliacinių duomenų bazių. Jį sukūrė originalūs MySQL kūrėjai ir garantuoja, kad ji išliks atviro kodo. Tai yra dažniausiai siūloma Linux duomenų bazė. MariaDB yra pagrįsta našumo, stabilumo ir atvirumo vertybėmis, o fondas užtikrina, kad indėlis bus priimtas atsižvelgiant į techninius nuopelnus. Naujausios naujos funkcijos apima pažangų grupavimą su Galera Cluster 4, suderinamumo su Oracle Database ir Temporal Data Tables funkcijas, leidžiančias užklausti duomenis, tokius, kokie jie buvo bet kuriuo momentu praeityje. MariaDB platinamas pagal **GPL** licenciją, 2 versiją. MariaDB klientų bibliotekos, skirtos C, Java ir ODBC, platinamos pagal LGPL licenciją, 2.1 ar naujesnę versiją. LGPL licencija leidžia laisvai platinti šias MariaDB klientų bibliotekas naudojant bet kurią programą.

.NET Core

.NET Core yra nauja .NET Framework versija, kuri yra nemokama, atvirojo kodo, bendros paskirties kūrimo platforma, prižiūrima Microsoft. Tai kelių platformų sistema, veikianti Windows, MacOS ir Linux operacinėse sistemose. .NET Core Framework gali būti naudojamas kuriant įvairių tipų programas, tokias kaip mobilios, darbalaukio, žiniatinklio, debesies, daiktų interneto, mašininio mokymosi, mikropaslaugų programėlės. .NET Core sukurta nuo nulio, kad ji būtų modulinė, lengva, greita ir kelių platformų sistema. Jame yra pagrindinės funkcijos, reikalingos norint paleisti pagrindinę .NET Core programą. Kitos funkcijos pateikiamos kaip „NuGet“ paketai, kuriuos prirėkus galite įtraukti į savo programą. Tokiu būdu .NET Core programa pagreitina našumą, sumažina atminties kiekį ir tampa lengvai prižiūrima. Ši programinė įranga platinama pagal **MIT** licenciją.

EntityFramework

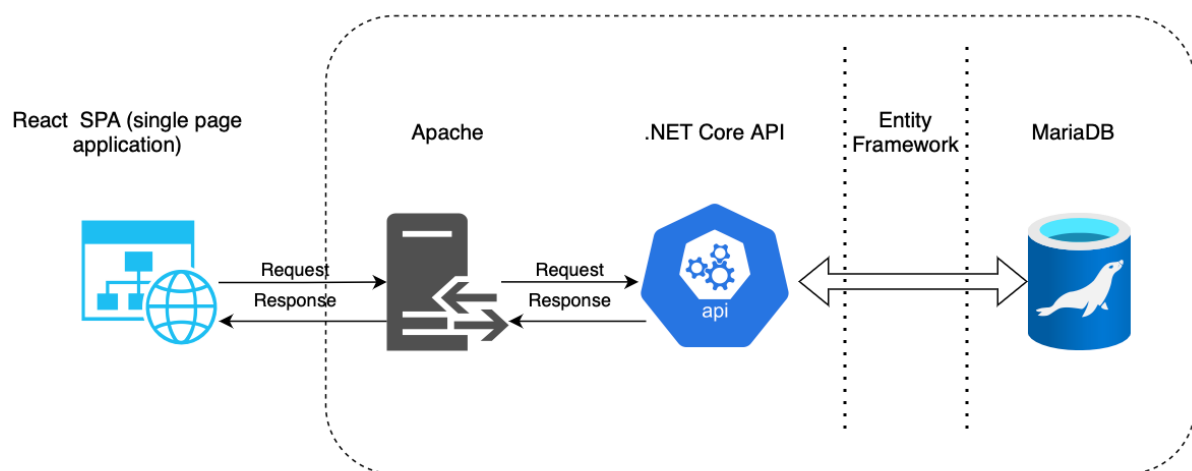
Entity Framework yra atvirojo kodo ORM sistema, skirta .NET programoms, kurias palaiko Microsoft. Tai leidžia kūrėjams dirbti su duomenimis naudojant specifinių domeno klasių objektus, nekreipiant dėmesio į pagrindines duomenų bazės lenteles ir stulpelius, kuriuose saugomi šie duomenys. Naudodami Entity Framework kūrėjai gali dirbti su aukštesniu abstrakcijos lygiu, kai dirba su duomenimis, ir gali kurti bei prižiūrėti į duomenis orientuotas programas naudodami mažiau kodo, palyginti su tradicinėmis programomis. Kaip ir .NET Core Entity Framework yra platinama **MIT** licencija.

React

React (taip pat žinomas kaip React.js arba ReactJS) yra nemokama atvirojo kodo JavaScript biblioteka, skirta kurti vartotojo sąsajas, pagrįstas UI komponentais. Jį prižiūri Meta (buvęs Facebook) ir atskirų kūrėjų bei įmonių bendruomenė. React gali būti naudojamas kaip pagrindas kuriant vieno puslapio ar mobiliąsias programas. Tačiau „React“ yra susijęs tik su būsenos valdymu ir tos būsenos pateikimu DOM, todėl kuriant React programas paprastai reikia naudoti papildomas bibliotekas maršruto parinkimui, taip pat tam tikras kliento funkcijas. 2017 m. rugsėjį Facebook paskelbė, kad React taps MIT licencijuota. Tai tapo realybe 2018 m. vasario 16 d., o tai reiškia, kad dabar visi su React susiję produktai dabar turi **MIT** licenciją.

Sistemos struktūrinė diagrama

Paveikslėlyje žemiau pateikta namų darbo sistemos diagrama. Sistemą sudaro 5 pagrindinės dalys. Duomenys bus įrašomi į MariaDB duomenų bazę iš jos duomenis nuskaityti arba įrašyti turi prieigą API. Tai padaryti padeda Entity Framework. Klientui duomenis atvaizduoti padeda naršyklės programėlė kuri parašyta su React ir siunčia HTML užklausas į Apache kuri veikia kaip reverse proxy ir perduoda jas API.



Gautas rezultas

API gražiną JSON failą su duomenimis

```
[{"id": "b8d38140-beb4-4eaa-a0d1-b609fd239e47",
  "firstName": "Ted",
  "lastName": "Mosby",
  "birthDate": "1995-11-29T00:55:21.88",
  "employmentDate": "2021-11-29T00:55:21.88",
  "bossId": "00000000-0000-0000-0000-000000000000",
  "homeAddress": "string",
  "currentSalary": 4500,
  "role": 0
}, {"id": "fe867a00-6294-4378-97c7-ca1b6ffc4cd5",
  "firstName": "Ted",
  "lastName": "Mosby",
  "birthDate": "1996-08-08T06:48:26.182",
  "employmentDate": "2021-11-16T02:32:42.276",
  "bossId": "b8d38140-beb4-4eaa-a0d1-b609fd239e47",
  "homeAddress": "Konstitucijos pr.",
  "currentSalary": 6500,
  "role": 2
}]
```

kurie paskui atvaizduojami lentelėje.

Employees

Firstname	Lastname	Salary	Role	Employment date	Home address
Ted	Mosby	4500 \$ 0		2021-11-29	string
Ted	Mosby	6500 \$ 2		2021-11-16	Konstitucijos pr.

Serverio konfigūravimo tvarka

1. Irasyti Apache

```
ds@ds: ~/Desktop
ds@ds:~/Desktop$ apache2 -v
Command 'apache2' not found, but can be installed with:
sudo apt install apache2-bin
ds@ds:~/Desktop$

ds@ds:~/Desktop$ sudo apt install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.2-0
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.2-0
0 upgraded, 9 newly installed, 0 to remove and 43 not upgraded.
Need to get 1,694 kB of archives.
After this operation, 7,169 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://lt.ports.ubuntu.com/ubuntu-ports focal/main arm64 libapr1 arm64 1.6
.5-1ubuntu1 [85.4 kB]
Get:2 http://lt.ports.ubuntu.com/ubuntu-ports focal/main arm64 libaprutil1 arm64
1.6.1-4ubuntu2 [82.4 kB]
Get:3 http://lt.ports.ubuntu.com/ubuntu-ports focal/main arm64 libaprutil1-dbd-s
qlite3 arm64 1.6.1-4ubuntu2 [10.4 kB]
Get:4 http://lt.ports.ubuntu.com/ubuntu-ports focal/main arm64 libaprutil1-ldap
arm64 1.6.1-4ubuntu2 [8,572 B]
```

2. Irasyti MariaDb

\$ sudo apt update

\$ sudo apt install mariadb-server

```
ds@ds: ~  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
43 packages can be upgraded. Run 'apt list --upgradable' to see them.  
ds@ds:~$ sudo apt install mariadb-server  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following additional packages will be installed:  
  galera-3 libcgi-fast-perl libcgi-pm-perl libconfig-inifiles-perl  
  libdbd-mysql-perl libdbi-perl libfcgi-perl libhtml-template-perl  
  libsnappy1v5 libterm-readkey-perl mariadb-client-10.3  
  mariadb-client-core-10.3 mariadb-common mariadb-server-10.3  
  mariadb-server-core-10.3 socat  
Suggested packages:  
  libclone-perl libmldbm-perl libnet-daemon-perl libsql-statement-perl  
  libipc-sharedcache-perl mailx mariadb-test tinycal
```

\$ sudo mysql_secure_installation

```
ds@ds: ~  
ds@ds:~$ sudo mysql_secure_installation  
  
NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB  
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!  
  
In order to log into MariaDB to secure it, we'll need the current  
password for the root user. If you've just installed MariaDB, and  
you haven't set the root password yet, the password will be blank,  
so you should just press enter here.  
  
Enter current password for root (enter for none):  
OK, successfully used password, moving on..  
  
Setting the root password ensures that nobody can log into the MariaDB  
root user without the proper authorisation.  
  
You already have a root password set, so you can safely answer 'n'.  
  
Change the root password? [Y/n] Y  
New password:  
Re-enter new password:  
Password updated successfully!  
Reloading privilege tables..  
... Success!  
  
By default, a MariaDB installation has an anonymous user, allowing anyone  
to log into MariaDB without having to have a user account created for  
them. This is intended only for testing, and to make the installation  
go a bit smoother. You should remove them before moving into a  
production environment.  
  
Remove anonymous users? [Y/n] n  
... skipping.
```

3. Sukonfiguruoti duomenu baze

```
ds@ds: ~  
ds@ds:~$ sudo mariadb  
Welcome to the MariaDB monitor. Commands end with ; or \g.  
Your MariaDB connection id is 63  
Server version: 10.3.31-MariaDB-0ubuntu0.20.04.1 Ubuntu 20.04  
  
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
MariaDB [(none)]> CREATE USER 'ubuntu'@'localhost' IDENTIFIED BY 'ubuntuuu';  
Query OK, 0 rows affected (0.003 sec)  
  
MariaDB [(none)]> GRANT ALL PRIVILEGES ON *.* TO 'ubuntu'@'localhost' WITH GRANT  
OPTION;  
Query OK, 0 rows affected (0.001 sec)  
  
MariaDB [(none)]> FLUSH PRIVILEGES;  
Query OK, 0 rows affected (0.001 sec)  
  
MariaDB [(none)]>
```

4. Irašyti ASP.NET Core Runtime, kad galima būtų paleisti API.

- a. Atsisiųsti Runtime paketą

```
ds@ds: /  
ds@ds:/$ sudo wget https://download.visualstudio.microsoft.com/download/pr/27840e8b-d61c-472d-8e11-c16784d40091/ae9780ccda4499405cf6f0924f6f036a/dotnet-sdk-5.0.100-linux-arm64.tar.gz  
--2021-12-05 13:56:03-- https://download.visualstudio.microsoft.com/download/pr/27840e8b-d61c-472d-8e11-c16784d40091/ae9780ccda4499405cf6f0924f6f036a/dotnet-sdk-5.0.100-linux-arm64.tar.gz  
Resolving download.visualstudio.microsoft.com (download.visualstudio.microsoft.com)... 68.232.34.200, 2606:2800:133:f17:19e8:2356:251b:2a9  
Connecting to download.visualstudio.microsoft.com (download.visualstudio.microsoft.com)|68.232.34.200|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 131895761 (126M) [application/octet-stream]  
Saving to: 'dotnet-sdk-5.0.100-linux-arm64.tar.gz.1'  
dotnet-sdk-5.0.100- 100%[=====] 125.79M  22.2MB/s   in 5.5s  
2021-12-05 13:56:08 (22.7 MB/s) - 'dotnet-sdk-5.0.100-linux-arm64.tar.gz.1' saved [131895761/131895761]  
ds@ds:/$
```

- b. Išpakakuoti persisųsta programą

```
ds@ds:/$ sudo tar xzf dotnet-sdk-5.0.100-linux-arm64.tar.gz -C $HOME/dotnet-64
```

- c. Iš eksportuoti paketus.

```
ds@ds:/$ export DOTNET_ROOT=$HOME/dotnet-64  
ds@ds:/$ export PATH=$HOME/dotnet-64:$PATH
```

- d. Galime patikrinti ar įrašyta įvedat komandą: `dotnet --info`

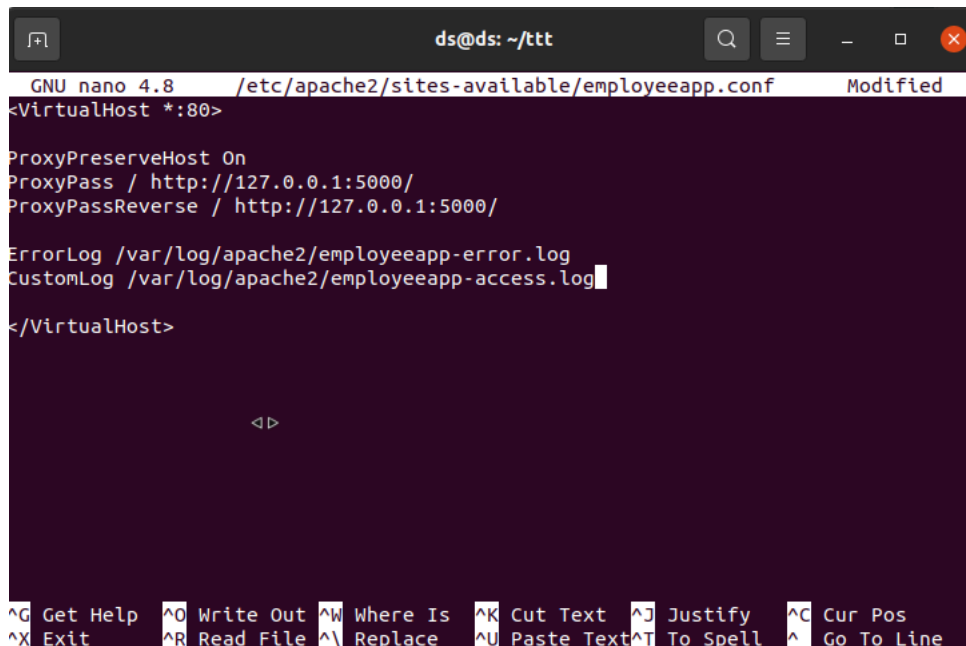
```
ds@ds:/$ dotnet --info  
.NET SDK (reflecting any global.json):  
Version: 5.0.100  
Commit: 5044b93829  
  
Runtime Environment:  
OS Name: ubuntu  
OS Version: 20.04  
OS Platform: Linux  
RID: ubuntu.20.04-arm64  
Base Path: /home/ds/dotnet-64/sdk/5.0.100/  
  
Host (useful for support):  
Version: 5.0.0  
Commit: cf258a14b7  
  
.NET SDKs installed:  
5.0.100 [/home/ds/dotnet-64/sdk]  
  
.NET runtimes installed:  
Microsoft.AspNetCore.App 5.0.0 [/home/ds/dotnet-64/shared/Microsoft.AspNetCore.App]  
Microsoft.NETCore.App 5.0.0 [/home/ds/dotnet-64/shared/Microsoft.NETCore.App]  
  
To install additional .NET runtimes or SDKs:  
https://aka.ms/dotnet-download
```

5. Sukonfiguruoti Apache serverį.

- a. Sukuriame naują .conf failą konfigūravimui.

`sudo nano /etc/apache2/sites-available/employeeapp.conf`

- b. Įr pridėdame eilutes pavaizduotas žemiau



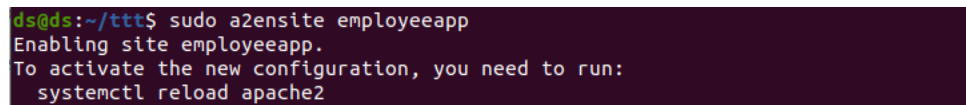
```
GNU nano 4.8 /etc/apache2/sites-available/employeeapp.conf Modified
<VirtualHost *:80>

ProxyPreserveHost On
ProxyPass / http://127.0.0.1:5000/
ProxyPassReverse / http://127.0.0.1:5000/

ErrorLog /var/log/apache2/employeeapp-error.log
CustomLog /var/log/apache2/employeeapp-access.log

</VirtualHost>
```

- c. Paskutinis žingsnis įgalinti ir perkrauti Apache



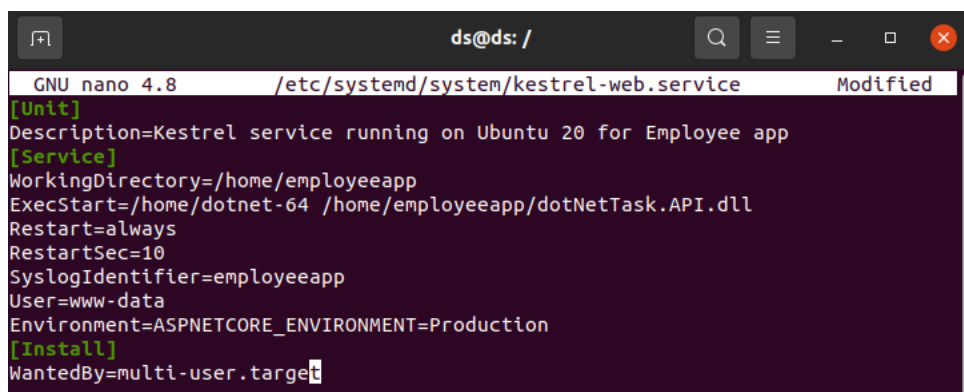
```
ds@ds:~/ttt$ sudo a2ensite employeeapp
Enabling site employeeapp.
To activate the new configuration, you need to run:
systemctl reload apache2
```

6. Įkeliame projekto failus (WEB API ir SPA) į Ubuntu. Paleidus build'ą „production“ režimui.
7. Paskutinis etapas užregistruoti kestrel servis'ą, kad programa pasileistu kas karta paleidus OS.

- a. Sukuriama .service konfigūracinį failą.

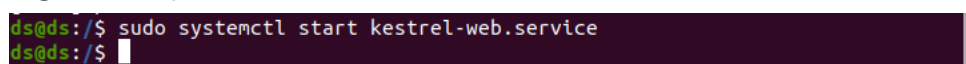
`$ sudo nano /etc/systemd/system/kestrel-web.service`

- b. Pridėdame eilutes pavaizduotas žemiau:



```
GNU nano 4.8 /etc/systemd/system/kestrel-web.service Modified
[Unit]
Description=Kestrel service running on Ubuntu 20 for Employee app
[Service]
WorkingDirectory=/home/employeeapp
ExecStart=/home/dotnet-64 /home/employeeapp/dotNetTask.API.dll
Restart=always
RestartSec=10
SyslogIdentifier=employeeapp
User=www-data
Environment=ASPNETCORE_ENVIRONMENT=Production
[Install]
WantedBy=multi-user.target
```

- c. Įr galiausiai paleisti servis'ą



```
ds@ds:/$ sudo systemctl start kestrel-web.service
ds@ds:/$
```

