**Гимназија Бора Станковић**

Вожда Карађорђа 27, Нип

**Матурски рад из базе података**

**Апликација намењена рецептима**

Ментор: Ученик:

Андрија Ђуришић, проф Никола Живковић IV-3

Ниш, јун 2022. год.

Садржај

[ОСНОВНИ ПОЈМОВИ 3](#_Toc104530362)

[.NET развојно окружење (.NET framework) 3](#_Toc104530363)

[Visual Studio 4](#_Toc104530364)

[C# 4](#_Toc104530365)

[Java Script 5](#_Toc104530366)

[Шта је JavaScript? 5](#_Toc104530367)

[Entity Framework (EF) 6](#_Toc104530368)

[Entity Framework - Code First Migration 7](#_Toc104530369)

[LINQ 8](#_Toc104530370)

[LINQ to SQL 9](#_Toc104530371)

[Апликација намењена раду са рецептима 10](#_Toc104530372)

[ТАБЕЛЕ АПЛИКАЦИЈЕ 12](#_Toc104530373)

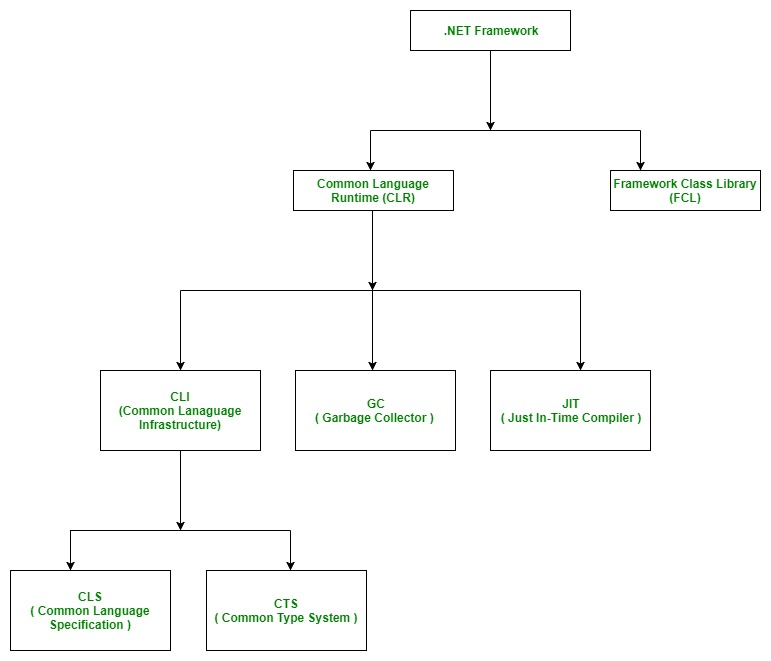
[ЛИТЕРАТУРА 13](#_Toc104530374)

# ОСНОВНИ ПОЈМОВИ

## .NET развојно окружење (.NET framework)

.NET је софтверско развојно окружење које је креирано и развијено од стране

Microsoft-a. Виртуална машина за компајлирање и извршавање програма писаних у различитим језицима као што су C#, VB.Net, итд. .NET се користи се за развој десктоп апликација, Web базираних апликација, и Web сервиса. Главне компоненте .NET развојног окружења су приказане на следећој слици:



* ***Common Language Runtime (CLR).*** Представља извршно окружење .NET развојног окружења које извршава кодове. Одговоран је за управљање извршењем .NET програма без обзира о ком .NET програмском језику се ради.
* ***Framework Class Library (FCL).*** Колекција библотека објектно-оријентисаних класа и метода, итд. која може бити интегрисана са CLR-ом. Инсталација .NET развојног окружења у основи представља инсталацију CLR-а и CLR-а у систему.
* ***Common Intermediate Language.*** Скуп инструкција које су независне од платформе. Компајлирањем изворног кода добија се код у том језику.
* ***Garbage Collector (GC)***. Озбезбеђује аутоматско ослобађање меморијског простора након што објекти више нису потребни.
* ***Just-In-Time (JIT) Compiler***. Део CLR-а који је задужен за извршавање .NET програма. Код у Common Intermediate Language-у се конвертује у машински код коришћењем овог компајлера. Машински код је зависан од рачунарског окружења на коме JIT компајлер ради.

## Visual Studio

Visual Studio је интегрисано развојно окружење развијено од стране Microsoft-а за развој више типова апликација: GUI (Graphicial User Interface) графичких корисничких апликација, конзолних апликација, Web апликација, апликација за мобилне уређаје, апликације у облаку, web сервисе, итд. Није предодређен програмски језк за који се користи. Могу се користити C#, C++, Visual Basic, Python, JavaScript, и многи други језици. Доступан је за Windows и macОS оперативне системе.

## C#

C# је објектно-оријентисан програмски језик креиран од стране Microsoft-а. Ради на .NET развојном окружењу. Развијен је од стране Адерс Хејлсберга и његовог тима унутар .NET иницијативе. Прва верзија је објављена 2002. године, а тренутно је актуелна верзија 8.0 из 2019. године.

Линк: https://www.geeksforgeeks.org/c-sharp-tutorial/

C# се користи за развој:

* мобилних апликација,
* десктоп апликација,
* Wеb апликација,
* Wеb сервиса,
* Wеb сајтова,
* игрица,
* апликација база података,
* итд.

Линк: <https://www.w3schools.com/cs/cs_intro.php>

## Java Script

JavaScript је најпопуларнији скрипт језик на интернету, и функционише на свим познатијим browser-има, као што су Chrome, Edge, Firefox, Opera, Internet Explorer, ...

### Шта је JavaScript?

* JavaScript је дизајниран да би се додала интерактивност HTML странама
* JavaScript је скрипт језик (једноставан програмски језик)
* JavaScript се састаји од линија кода који може да се извршава
* JavaScript се обично директно уграђује у HTML стране
* JavaScript је интерпретерски језик (што значи да се извршава без претходног компајлирања)
* JavaScript није лиценциран језик

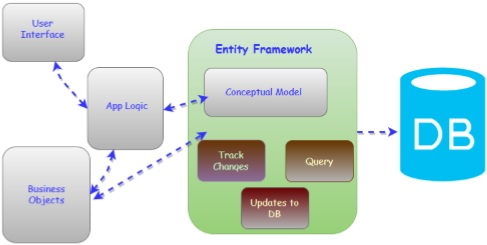
Извор: <http://es.elfak.ni.ac.rs/iw/Materijal/Vezbe/JavaScript.pdf>

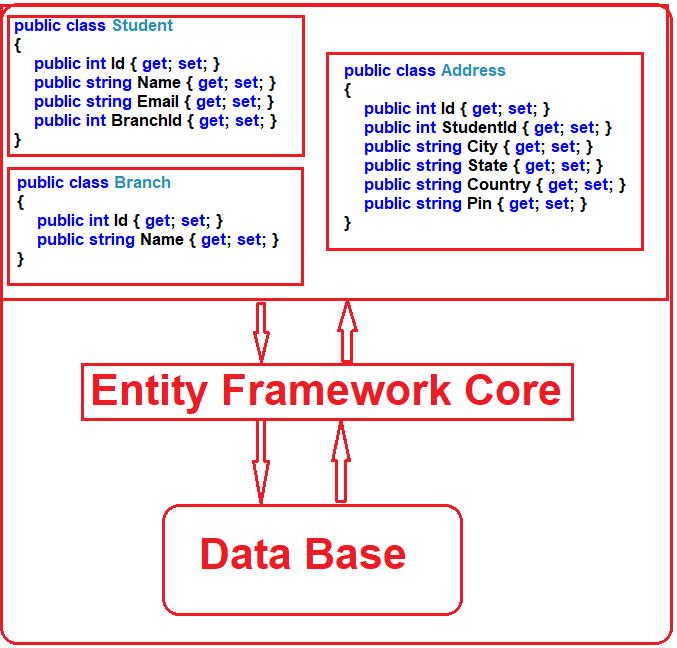
## Entity Framework (EF)

Entity framework (скраћеница, EF) је ORM развојно окружење. ORM је скраћеница од објектно-релационо мапирање. Ово радно окружење је део Microsoft .NET развојне платформе. Сврха EF-а је да апстрахује везе са релационом базом података, на такав начин да се апликација може повезати са ентитетом базе података као са скупом објеката. (Извор:https://www.c-sharpcorner.com/article/entity-framework-introduction-using-c-sharp-part-one/ ). EF је тип алата који поједностављује мапирање између објеката у софтверу и табела и колона релационе базе података.

Од стране Microsoft-а Entity Framework је препоручена технологија за начин приступа подацима за нове апликације.

Основни фокус EF-а је на објектном моделу саме апликације, а не на моделу базе података у којој се сами подаци апликације чувају. Могуће је да су концепутални модел апликације и модел базе података (шема базе података) у сагласности, али могу бити и крајње различити, тако да овај систем обезбеђује одговарајуће мапирање.

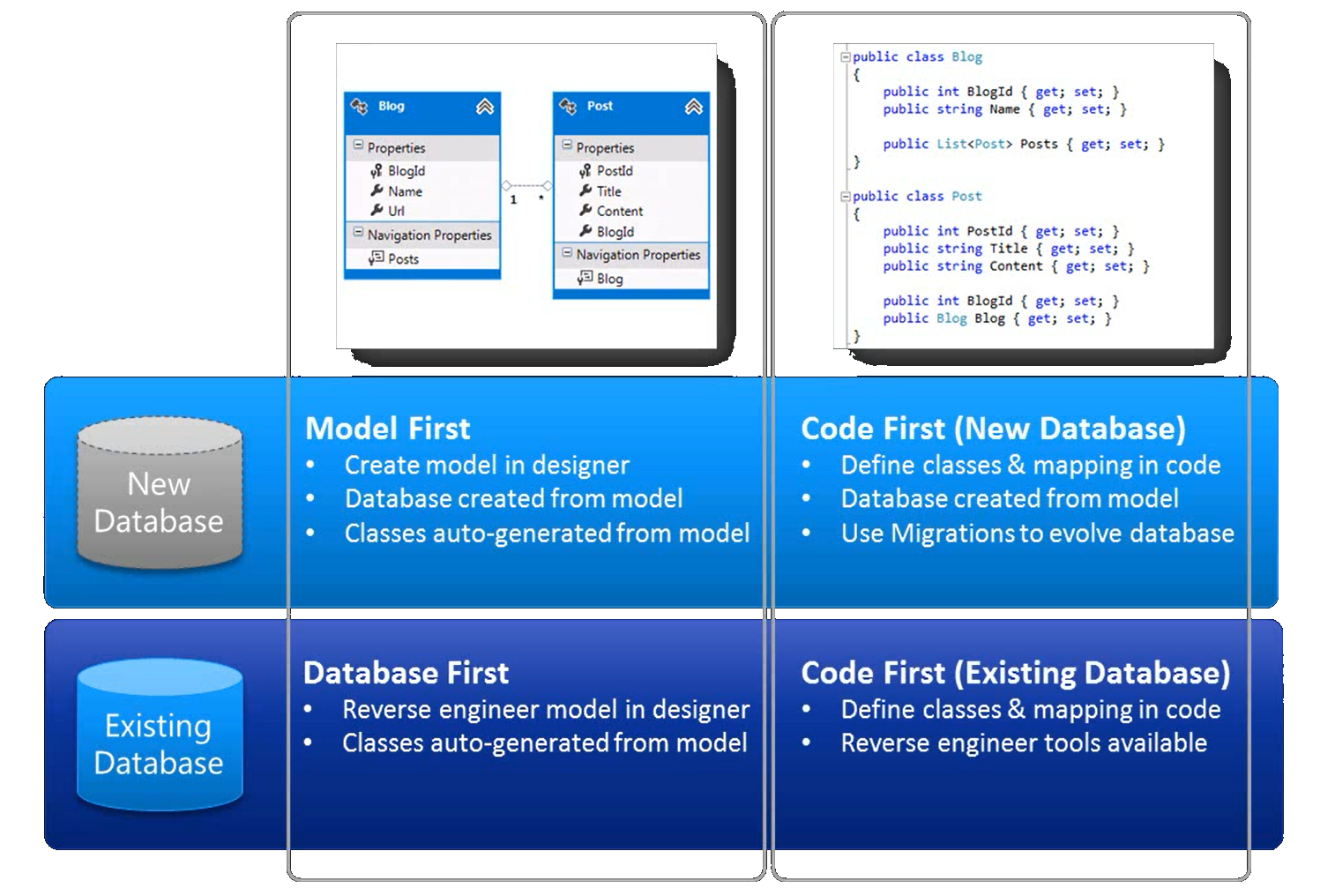




Entety Framework обезбеђује три приступа реализацији модела ентитета (модела базе података):

* Code First (прво код)
* Database First (прво база података) и
* Model First (прво модел).

На следећој слици су приказани случајеви у којима се користе поједини приступи, као и кратак опис поступка.

Детаљније: https://download.microsoft.com/download/8/F/8/8F81F4CD-3678-4229-8D79-0C63FFA3C595/HDI\_ITPro\_Technet\_winvideo\_ChoseYourWorkflow.wmv

### Entity Framework - Code First Migration

Entity Framework од верзије 4.3 обезбеђује нову функцију Code First Migration, која омогућава инкрементални развој шеме базе података, а како се модел мења током развоја апликације. Са миграцијом, он ће аутоматски ажурирати шему базе података, када се модел претрпи измену без губутка постојећих података и објеката базе података. Користи се нови иницијализатор који се назива *MigrateDatabaseToLatestVersion*.

Постоје две врсте миграције:

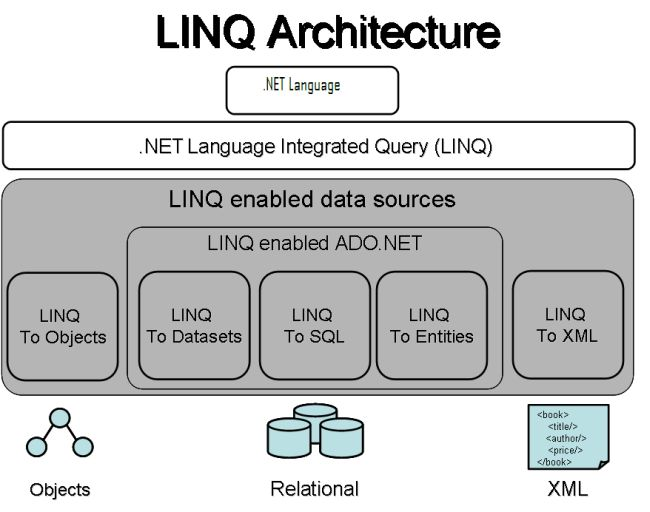
* Аутоматизована миграција и
* Код базирана миграција.

Детаљнији опис је доступан на следећем линку:

<https://www.tutorialspoint.com/entity_framework/entity_framework_code_first_migration.htm>

## LINQ

**LINQ** је скраћеница од **L**anguage **IN**tegrated **Q**uery (упит интегрисан у језику) и углавном се користи за упите над подацима (упис и преузимање података) из различитих извора података као што су SQL базе података, XML документи, ADO.NET Dataset-ови, .NET колекције, итд. Први пут је уведен у .NET 3.5.

****

Извор: <https://www.c-sharpcorner.com/UploadFile/84c85b/using-linq-with-C-Sharp-2012/>

Као одговор на потребу да постоји јединствени и заједнички језик упита који ради са различитим изворима података, настао је LINQ. Потребно је научити LINQ и тиме обезбедити могућност рада са било којим типом извора података.

LINQ провајдери, попут LINQ то Објецтс, LINQ то Ентитиес, LINQ то XML итд. су компоненте између LINQ упита и извора података које конвертују LINQ операције у специфичне команде које извори података могу разумети. На пример LINQ то SQL провајдер конвертује LINQ упит у T-SQL тако да базе података могу разумеит исте.

Извор:https://www.manuelradovanovic.com/2018/04/uvod-u-linq.html

Ламбда изрази се опционо користе уместо именованих метода и поједностављују кодирање. Ламбда израз је кратак и концизан начин писања анонимних метода, значи метода који немају име, који се користе за извршавање упита. Ламбда оператор се пише овако =>.

### LINQ to SQL

LINQ to SQL је могућност управљања и приступања релационим подацима као објектима. Он се повезује на базу података, конвертује LINQ конструкције у SQL, прослеђује SQL, трансформише резултате у објекте, чак прати измене и аутоматски захтева ажурирање базе података.

Једноставан LINQ упит захтева три ствари: ентитy класу, контекст података и LINQ упит.

Линк: [https://www.c-sharpcorner.com/UploadFile/84c85b/using-linq-with-C-Sharp-2012/#:~:text=LINQ%20to%20Objects-,Using%20LINQ%20to%20SQL,-LINQ%20to%20SQL](https://www.c-sharpcorner.com/UploadFile/84c85b/using-linq-with-C-Sharp-2012/" \l ":~:text=LINQ to Objects-,Using LINQ to SQL,-LINQ to SQL)

// Specify the data source.

int[] scores = { 97, 92, 81, 60 };

// Define the query expression.

IEnumerable<int> scoreQuery =

from score in scores

where score > 80

select score;

// Execute the query.

foreach (int i in scoreQuery)

{

Console.Write(i + " ");

}

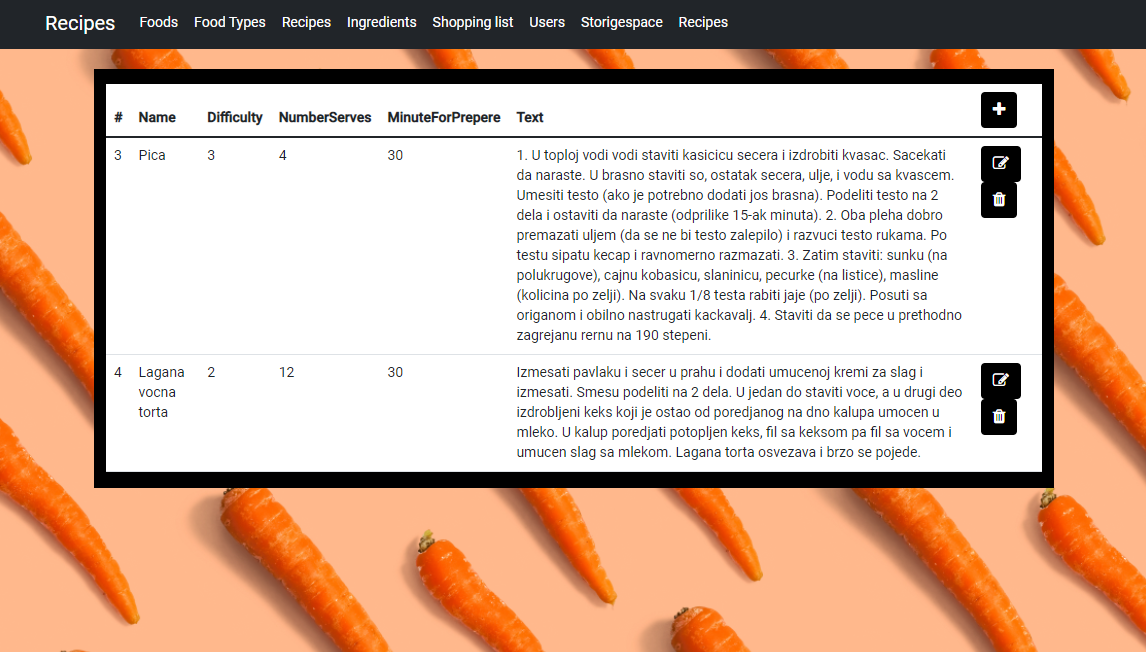
// Output: 97 92 81

# Апликација намењена раду са рецептима

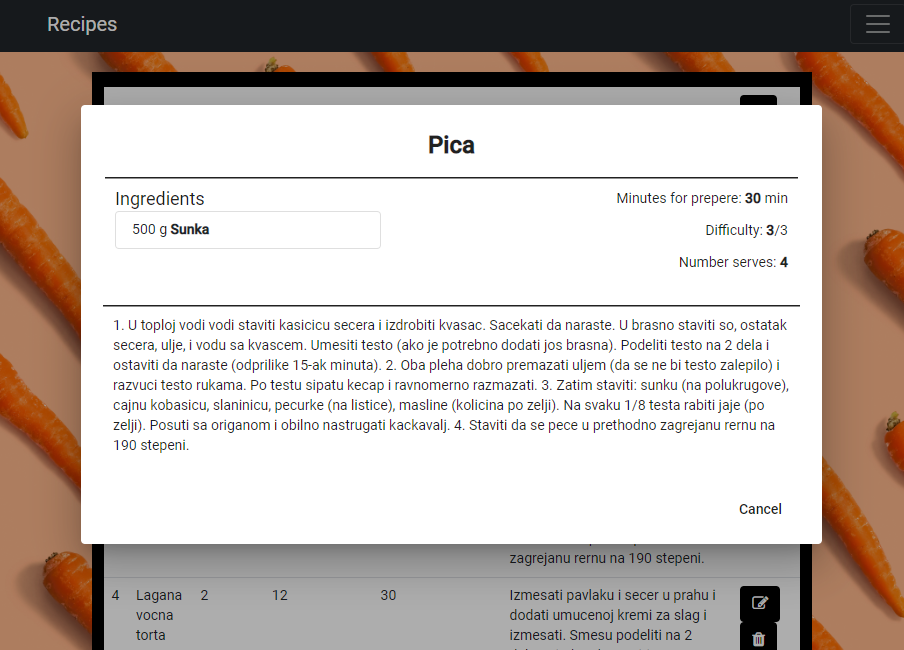
Веб апликација намењена је раду са рецептима. Сваки рецепт садржи намирнице које се користе за израду јела, као и опис поступка израде. Намирнице су подељене по типовима намирница. Такође се у бази чува и место где су намирнице смештене.

Приступом станици за рад са рецептурама приказује се основни екран дат на следећој слици, а на коме су приказани унети рецепти.

На врху екрана налази се навигациони бар, преко кода се долази до појединих страница веб апликације.

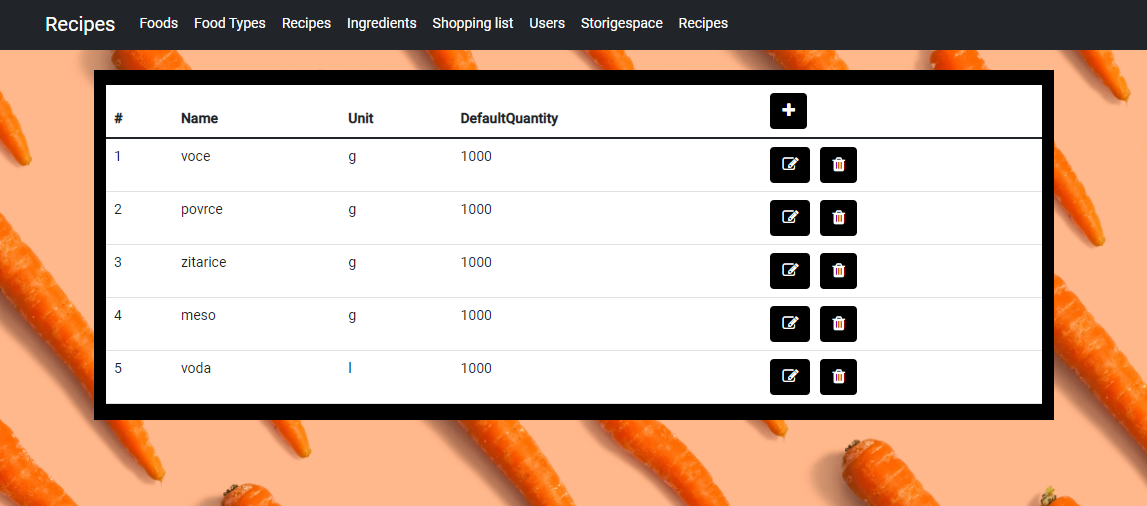


Притиском на изабрани рецепт појављује се следећи искачући прозор за приказ изабране рецептуре.



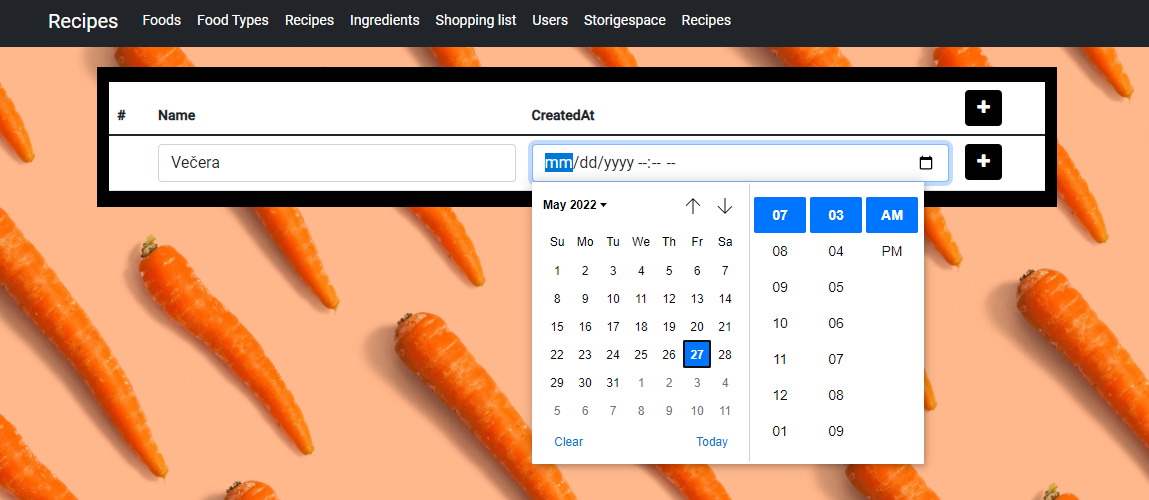
Притиском на “Foods“ приказује се екран на коме се уносе и ажурирају сви типови намирница који ће бити коришћени у изради рецепата.

Притиском на „Food Types“ приказује се екран типова намирница. Подела по типовима намирница је уведена ради каснијег лакшег претраживања намирница код израде саме рецептуре.

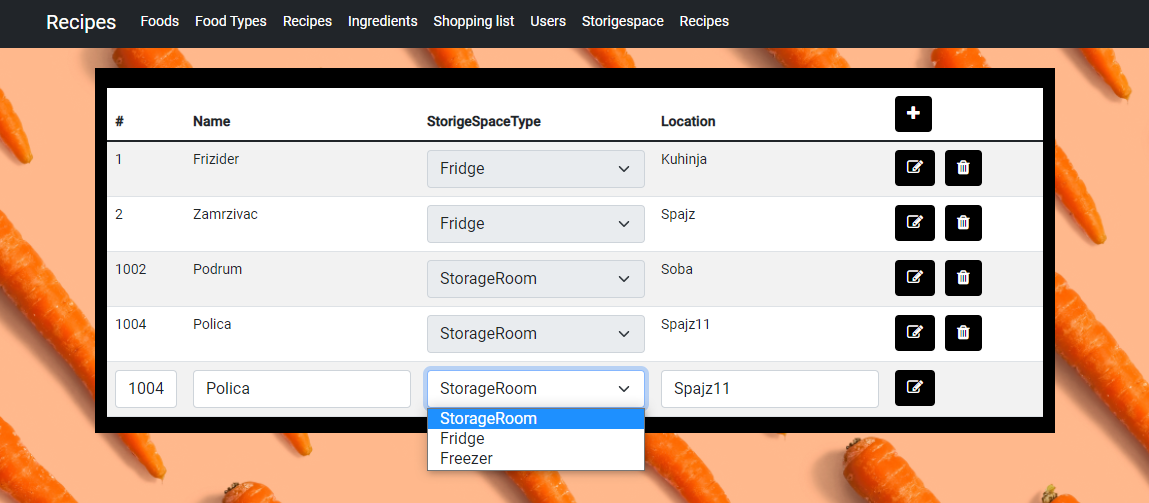


Приском на “Ingredients” приказује се екран на коме се уноси састав одређене рецептуре.

Притиском на “Shopping list” приказује се екран на коме се креира листа за куповину.



Притиском на „Storagespace“ приказује се екран на коме можемо додати просторе на којима чувамо намирнице.



Притиском на „Users“ приказује се екран на коме додајемо кориснике.

## ТАБЕЛЕ АПЛИКАЦИЈЕ

На следећој слици су пиказане табле апликације.

|  |  |
| --- | --- |
| Табела | Опис |
| Foods | Табела намирница које се користе у изради рецептура |
| FoodTypes | Типови намирница |
| Recipes | Табела рецептура |
| Ingredients | Табела намирница које се користе у конкретној рецептури |
| ShoppingLists | Табела листе за куповину |
| ShoppingListItems | Табела листе намирница листе за куповину |
| StorageSpaces | Место чувања намирница |
| UsedRecipes | Табела листе коришћених рецептура |
| StoredMaterials | Табела употребљених намирница |
| Users | Листа корисника |

# ЛИТЕРАТУРА

* <https://www.geeksforgeeks.org/c-sharp-tutorial/>
* <https://www.w3schools.com/cs/cs_intro.php>
* <http://es.elfak.ni.ac.rs/iw/Materijal/Vezbe/JavaScript.pdf>
* <https://www.c-sharpcorner.com/article/entity-framework-introduction-using-c-sharp-part-one>
* https://download.microsoft.com/download/8/F/8/8F81F4CD-3678-4229-8D79-0C63FFA3C595/HDI\_ITPro\_Technet\_winvideo\_ChoseYourWorkflow.wmv
* <https://www.tutorialspoint.com/entity_framework/entity_framework_code_first_migration.htm>
* <https://www.c-sharpcorner.com/UploadFile/84c85b/using-linq-with-C-Sharp-2012/>
* <https://www.manuelradovanovic.com/2018/04/uvod-u-linq.html>
* [https://www.c-sharpcorner.com/UploadFile/84c85b/using-linq-with-C-Sharp-2012/#:~:text=LINQ%20to%20Objects-,Using%20LINQ%20to%20SQL,-LINQ%20to%20SQL](https://www.c-sharpcorner.com/UploadFile/84c85b/using-linq-with-C-Sharp-2012/#:~:text=LINQ to Objects-,Using LINQ to SQL,-LINQ to SQL)