

## DESIGNING A FOOD ORDERING WEBSITE FROM MULTIPLE REGISTERED RESTAURANTS

*Assoc. Prof. PhD Dimitar Minchev, Burgas Free University, [mitko@bfu.bg](mailto:mitko@bfu.bg)  
Hristo Gemedjiev, graduate student, [18311015@students.bfu.bg](mailto:18311015@students.bfu.bg)*

**Abstract:** The purpose of the provided content is to present the process of creating a dynamic web site for restaurant registration and their menus, as well as customer ordering using mainly Python's Django framework. The processes of creating the site are presented, namely: Creation of a concept for the construction of the site; Selection of tools for creating the site; Construction and administration of the site.

**Keywords:** e-commerce, online shopping, new age restaurants.

## ПРОЕКТИРАНЕ НА УЕБ САЙТ ЗА ПОРЪЧКА НА ХРАНА ОТ МНОЖЕСТВО РЕГИСТРИРАНИ РЕСТОРАНТИ

*доц. д-р Димитър Минчев, Бургаски свободен университет, [mitko@bfu.bg](mailto:mitko@bfu.bg)  
Христо Гемеджиев, дипломант, [18311015@students.bfu.bg](mailto:18311015@students.bfu.bg)*

**Абстракт:** Целта на предоставеното съдържание е да се представи процеса на създаване на динамичен уеб сайт за регистрация на ресторанти и техните менюта, както и поръчка от страна на клиенти използвайки основно Django framework на Python. Представени са процесите на създаване на сайта, а именно :Създаване на концепция за изграждането на сайта; Подбор на инструменти за създаването на сайта; Изграждане и администриране на сайта.

**Ключови думи:** e-commerce, online shopping, new age restaurants.

### Въведение

С настъпилият Ковид-19 и противоепидемичните мерки, ресторантският бизнес изгуби доста клиенти. Много от по-малките ресторанти не успяха да превъзможнат мерките за борба с корона вируса и фалираха, което своевременно води и до загуби за държавният бюджет. Поради тази причина сайтове като „New Age Restaurants“, могат да спасят малките ресторанти, както и да подпомогнат по-големите вериги ресторанти да увеличат своя оборот. В днешно време всеки човек е запознат с електронните технологии и има смартфон в себе си както и компютър или лаптоп вкъщи. В свободното си време повечето хора си „ровят“ в мрежата, където вече навсякъде има реклами на определени продукти и сайтове, които предлагат своите продукти или услуги. С две думи, мнозинството от хора знаят как да използват онлайн сайтове и как да закупят продукт или услуга онлайн. Този факт води до постоянното увеличаване бройката на подобни сайтове. Реално погледнато в България е наложен монопол върху сайтовете за разнос на храна. Тези сайтове са: „Takeaway“ и „Glovo“. Всичко от страна на клиента изглежда страотно, все пак можеш да си поръчаш храна от вкъщи, да не се занимаваш с пазаруване, готвене, чистене и в същото време да ти я донесат до вратата, всичко това за около 2-3 лв. доставка. Но тъжната

истина е, че от всичко това губи самият ресторант. Собственик на ресторант, който работи с гореспоменатите доставчици и всъщност те взимат тези 2-3 лв. такса за разнос и в същото време взимат немалък процент от цената на самите ястия, тоест ресторантите приготвят ястията с почти никаква реална печалба от това, което за големите вериги може би не е толкова тежко, но малките ресторанти се „пръскат от работа“ и в същото време отчитат минимални или нулеви печалби от онлайн продажбата на ястия. Този собственик на ресторант вече не работи с доставчиците на храна. Създаде личен сайт на ресторанта, от който можеш да си поръчаш храна. Нае си служител, който разнася единствено за неговият ресторант, с което е увеличил драстично печалбата на ресторанта си.

## СЪЗДАВАНЕ НА КОНЦЕПЦИЯ ЗА РАЗРАБОТКА НА САЙТ ЗА ОНЛАЙН ПОРЪЧКА НА ХРАНА.

Най-важната стъпка за изработката на успешен сайт е създаването на концепция. За да се случи това, първо трябва да отговорим на следните въпроси:

### ЦЕЛ НА САЙТА

Целта на сайта „New Age Restaurants“ е да се поръча лесно и бързо храна за вкъщи от любимите ти ресторанти покрай теб. Храната ще бъде доставяна с лични доставчици на регистрираните ресторанти. Ресторантите ще могат лесно собственооръчно да направят заявка за регистриране на техният ресторант на сайта. След като заявката бъде разгледана от администратора на сайта, ресторантът ще бъде одобрен и регистрацията му ще бъде активирана и съответно ще получи имейл, че е одобрен. След това ще може да добави своето меню, като ще може да създаде свои категории както и ястия.

### ЦЕЛЕВА ГРУПА

Сайтът е насочен както към клиенти, които да си поръчат храна така и към ресторанти, които искат да се включат към платформата или да избягат от големите такси на монопола в тази сфера. Ресторантите, които вече имат собствени сайтове от където клиентите могат да поръчват, също ще са доволни, защото вече няма да е нужно да плащат на „developer“ да поддържа сайтът им, както и няма да плащат отделно такса за хостинг.

### СЪДЪРЖАНИЕ НА САЙТА

Съдържанието на сайта ще бъде основно от различни ресторанти, с техните менюта, работно време и локация.

Ресторантите ще имат форма, чрез която да направят заявка за включване към сайта, след което заявката ще бъде прегледана от администраторът на сайта и евентуално одобрена. След одобряването, ресторантът ще може да се „логне“ в сайта, където в своето табло ще може да въведе информация, като снимки(лого, снимка за фон на ресторантската страница), адрес – чрез „Google Places Api“, тоест ако вече има зададена точка на Google Maps, ще може директно да я избере, за да може клиентите лесно и бързо да го намират. След това ще може да създаде своето меню, като първо започне с категориите(предястие, основно, рибни ястия, напитки, десерт и т.н.). После към всяка категория ще може да добавя ястия като въведе име, цена, описание и снимка на ястието.

От клиентската страна, те ще могат да разглеждат целият сайт, но за да поръчат от даден ресторант първо ще трябва да се регистрират в сайта. След като попълнят формата за регистрацията, клиентите ще получат имейл с линк за потвърждение на регистрацията, след като го кликнат, акаунтът им ще се активира и вече ще могат да се „логнат“. След като са влезли в профила си, те също имат свое табло където могат да въведат своя адрес, отново чрез Google Places. Ще могат да си сложат профилна снимка и корица. Имат и прозорец където ще са запазени всички техни поръчки до сега и за всяка ще могат да видят детайли, като – дата, номер на поръчка, начин на плащане, какво са поръчали и т.н.

### РЕЗУЛТАТ ОТ СЪЗДАВАНЕТО НА САЙТА

Сайтът е изграден с комерсиална цел, тоест всеки ресторант ще трябва да плаща малка месечна такса, за да може ресторантът им да бъде активен на сайта. Таксата ще бъде малка за да може да се привлекат възможно най-много ресторанти, дори и малките, които не могат да си позволят да плащат големи такси. Цялостната идея е да отървем ресторантите, в специалност малките, от здравата хватка на монопола в тази сфера и да им предложим по-добри условия с по-млаки такси.

## ИЗБОР НА ИНТРУМЕНТИ ЗА СЪЗДАВАНЕ НА САЙТА

Тъй като разнообразието от инструменти за създаването на уеб сайт е огромно, ще трябва да изберем точно кои технологии да ползваме.

Избиране на софтуер за създаване на сайта – Django Web Framework (Python)

Първата стъпка при създаването на един сайт е избирането на подходящите технологии. Django е Python Web Framework от високо ниво, която позволява бързо разработване на сигурни и лесно поддържани уебсайтове. Създаден от опитни разработчици, Django се грижи за голяма част от трудностите при уеб разработката, така че можете да се съсредоточите върху писането на вашето приложение, без да е необходимо да преоткривате колелото. Той е безплатен и с отворен код, има процъфтяваща и активна общност, страхотна документация и много опции за безплатна и платена поддръжка.

Django помага да пишем софтуер, който е:

**Завършен**

Django следва философията „Включени батерии“ и предоставя почти всичко, което разработчиците биха искали да направят „извън кутията“. Тъй като всичко, от което се нуждаете, е част от един „продукт“, всичко работи безпроблемно заедно, следва последователни принципи на проектиране и има обширна и актуална документация.

**Разнообразен**

Django може да се използва (и е бил) използван за изграждане на почти всякакъв вид уебсайтове - от системи за управление на съдържанието и „wiki“, до социални мрежи и новинарски сайтове. Може да работи с всяка клиентска рамка и може да доставя съдържание в почти всеки формат (включително HTML, RSS емисии, JSON и XML).

Въпреки че предоставя избор за почти всяка функционалност, която бихте искали (напр. няколко популярни бази данни, модули за шаблони и т.н.), той може също да бъде разширен, за да използва други компоненти, ако е необходимо.

**Сигурен**

Django помага на разработчиците да избегнат много често срещани грешки в сигурността, като предоставя рамка, която е проектирана да "прави правилните неща", за да защити уебсайта автоматично. Например Django предоставя сигурен начин за управление на потребителските акаунти и пароли, като избягва често срещани грешки като поставяне на информация за сесията в бисквитките, където е уязвима (вместо това бисквитките просто съдържат ключ и действителните данни се съхраняват в базата данни) или директно съхраняване на пароли а не хеш парола.

Хешът на паролата е стойност с фиксирана дължина, създадена чрез изпращане на паролата чрез криптографска хеш функция. Django може да провери дали въведената парола е правилна, като я пусне през хеш функцията и сравни изхода със съхранената хеш стойност. Въпреки това, поради „еднопосочния“ характер на функцията, дори ако съхранената хеш стойност е компрометирана, за атакуващия е трудно да разработи оригиналната парола.

Django позволява защита срещу много уязвимости по подразбиране, включително SQL инжектиране, междусайтови скриптове, фалшифициране на междусайтови заявки и „clickjacking“.

**Мащабируем**

Django използва базирана на компоненти архитектура - „споделено нищо“ (всяка част от архитектурата е независима от другите и следователно може да бъде заменена или променена, ако е необходимо). Наличието на ясно разделение между различните части означава, че може да се мащабира за увеличен трафик чрез добавяне на хардуер на всяко ниво: кеширащи сървъри, сървъри на бази данни или сървъри на приложения. Някои от най-натоварените сайтове успешно са мащабирали Django, за да отговорят на техните изисквания (напр. Instagram и Disqus).

**Поддържа се**

Кодът на Django е написан с помощта на принципи и модели на дизайн, които насърчават създаването на лесен за поддръжка и многократно използваем код. По-специално, той използва принципа „Не се повтаряй“ (DRY), така че няма ненужно дублиране, намалявайки количеството код. Django също насърчава групирането на свързана функционалност в „приложения“ за

многократна употреба и на по-ниско ниво групира свързания код в модули – „Model View Controller“(MVC).

Преносим

Django е написан на Python, който работи на много платформи. Това означава, че не сте обвързани с никоя конкретна сървърна платформа и можете да стартирате приложенията си на много версии на Linux, Windows и macOS. Освен това Django се поддържа добре от много доставчици на уеб хостинг, които често предоставят специфична инфраструктура и документация за хостване на сайтове на Django.

Django

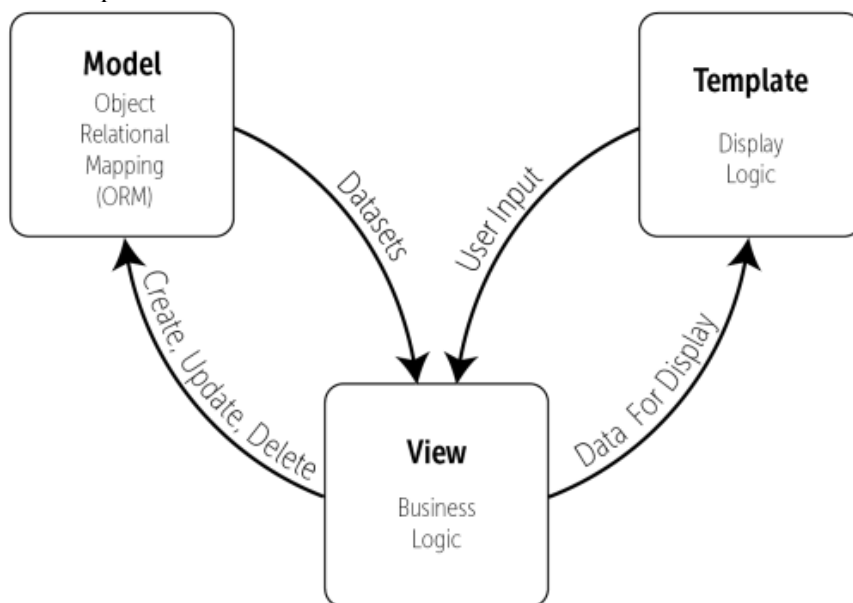
На най-високо ниво Django е Model-View-Controller – или MVC – фреймуърк. MVC е модел за проектиране на софтуер, който има за цел да раздели уеб приложение на три взаимосвързани части:

- **Моделът (Model)**, който предоставя интерфейса с базата данни, съдържаща данните на приложението;
- **Изгледът (View)**, който решава каква информация да бъде представена на потребителя и събира информация от потребителя;
- **Контролерът (Controller)**, който управлява бизнес логиката за приложението и действа като информационен посредник между модела и изгледа.

Django използва малко по-различна терминология в своята реализация на MVC (Фиг. 1).

В Django:

- **Моделът** функционално е същият. Обектно-реляционното картографиране на Django (ORM) предоставя интерфейсът към базата данни на приложението;
- **Темплейтът (Template)** предоставя външната логика и е интерфейсът между потребителя и нашето Django приложение;
- **Изгледът** управлява по-голямата част от обработката на данни на приложенията, логиката на приложенията и съобщенията.



Фиг. 1. Описание на MTV шаблонът на Django

## ИЗГРАЖДАНЕ И АДМИНИСТРИРАНЕ НА САЙТА.

Избиране на среда за програмиране(IDE)

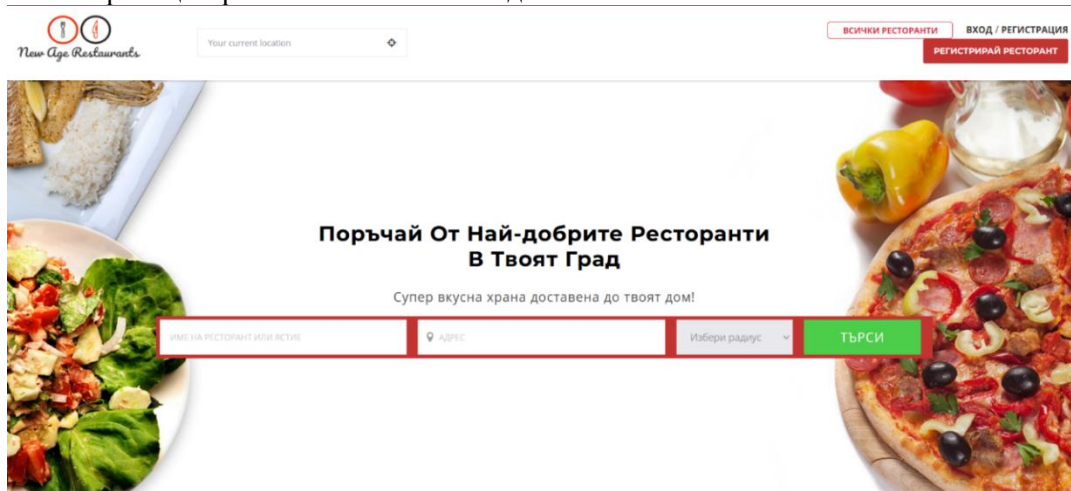
Първото нещо, което трябва да направим преди да започнем изграждането на сайта е да изберем среда за програмиране. Тъй като ще използваме езикът Python, затова избираме най-използваната среда за програмиране на Python – PyCharm Professional. Тази версия е платена, но понеже все още съм студент използвам безплатният студентски лиценз за да я ползвам. Има и безплатна версия – PyCharm Community Edition, но тя е доста оръзана от към функции.

### Създаване на проект в PyCharm

След като отворим PyCharm Professional, натискаме върху менюто „File“ и избираме „New Project“. От ново изсочилият прозорец, натискаме на опцията Django(PyCharm Professional има вградена поддръжка и специфични функции, които подпомагат и улесняват програмирането с Django), избираме име на проекта и натискаме „Create“. PyCharm автоматично създава и виртуална среда за програмиране (Virtual Environment). Което е задължително, защото когато започне да инсталираме Python модули като: Django, Pillow и т.н., ние не искаме да ги инсталираме директно на глобалната машина, защото ще се „задръсти“ от твърде много модули, които най-вероятно няма да използваме на всичките си проекти. Затова се създава виртуална среда на която ще се инсталират определените модули, които ще ни трябват за проекта. Вече празният ни проектът е създаден, PyCharm автоматично е инсталирал модульът Django и предвождащите го малки модули.

### Потребителски интерфейс

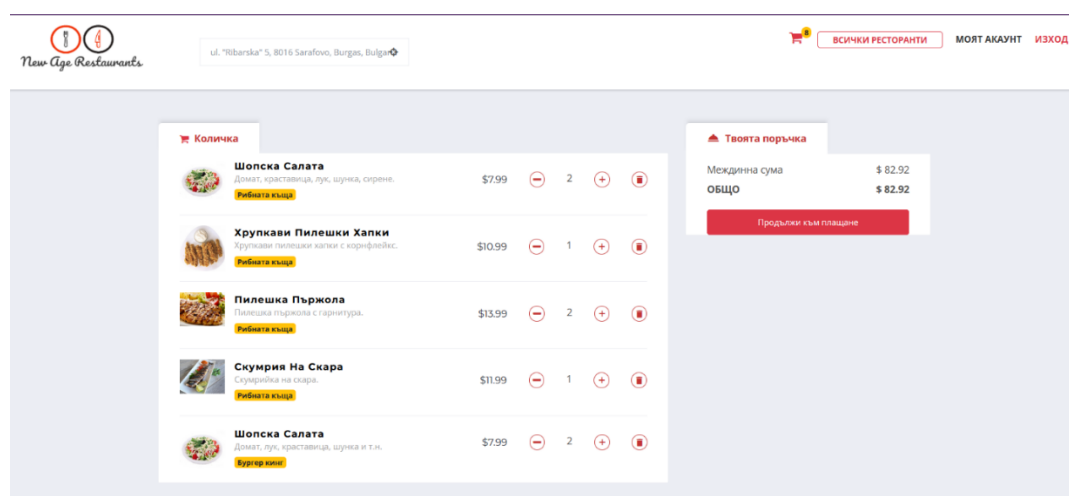
Началната страница е разположена както следва:



Фиг. 2. Началната страница

### Количка

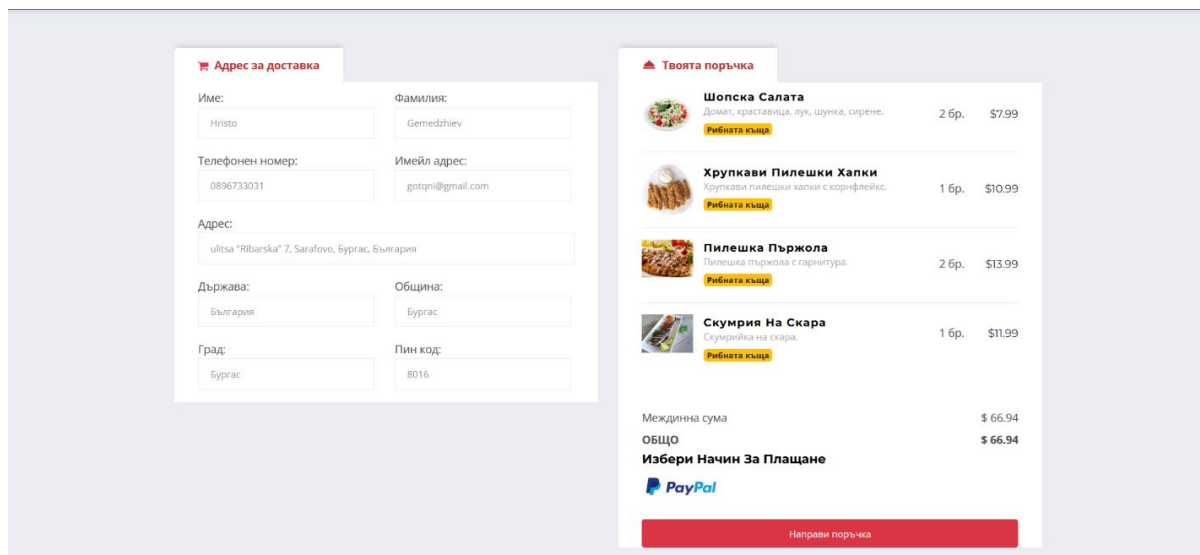
В количката се намират всички продукти, които потребителят е добавил. Ако не са добавени никакви продукти, на мястото на продуктите се изписва „Количката е празна“. За всеки продукт се изписва информация както следва: име на ястието, описание, цена, от кой ресторант е самото ястие и брой. Броят може да се коригира съответно с бутоните „+“ и „-“, както и да се премахне определеното ястие с бутона „кошче“. Отдясно се изписва междинната сума и общата сума както и бутонът „Продължи към плащане“.



Фиг. 3. Количка

### Страница Checkout

Отляво е адресът за доставка, както и информация за клиента. Информацията в тези полета е автоматично попълнена, понеже потребителят вече е въвел тази информация в своя профил и тук може да я коригира ако например в този момент се намира на друга локация или поръчва за свой приятел. Отдясно е отново резюме на избраните от потребителят продукти, както и цената. Има опция да се избере по какъв начин клиентът иска да плати поръчката. За моментът, единственият избор е PayPal (с него може да се плати чрез регистрацията или чрез кредитна/дебитна карта), но след като проектът се развие повече ще се добави и опция „наложен платеж“. След като изберем начинът на плащане (в случаят PayPal), се отваря страницата на която вече избираме дали да платим от профил в PayPal или чрез карта.



Фиг. 4. Страница Checkout

### Заклучение

С помощта на Django е изработен сайт за онлайн поръчки на храна за офиса и дома. Чрез него лесно и бързо можеш да избереш ресторант, който е най-близо до теб, само с кликането на един бутон. В сайтът потребителите лесно могат да създадат своя профил, както и да го персонализират със собствена корица и профилна снимка. Също така лесно и бързо могат да изберат своя адрес с помощта на Google Places API. Когато си поръчват продукти, адресът им е автоматично въведен, което допълнително улеснява тяхното ползване на сайта. В своето табло те имат детайлна таблица с предишните си поръчки, която могат да използват ако например искат да си следят разходите или пък ако са забравили, кое беше онова вкусно ястие, което ядоха преди 1 седмица. Ресторантите могат с минимална такса да получават допълнителни поръчки онлайн, които сами да менажират, а не да чакат доставчик, който има още 15 поръчки да предаде, преди да вземе тяхното ястие, което вече изстива. Те лесно могат да се регистрират в сайта, след което да започнат да създават своето меню, което става лесно и бързо. Те сами определят кои ястия да се продават онлайн и кои дни да работят с доставки. Администраторът на сайта може лесно и бързо да поправи грешки в базата данни, без да спира сайта за поддръжка.

### Използвана литература

- [1] <https://masteringdjango.com/django-tutorial-for-beginners/>
- [2] <https://aiven.io/blog/an-introduction-to-postgresql/>
- [3] <https://docs.djangoproject.com/en/4.1/>
- [4] <https://docs.djangoproject.com/en/4.1/topics/signals/>