

РАЗВОЈ ВЕБ АПЛИКАЦИЈЕ ТИПА MASTER-DETAIL – ПОТЕНЦИЈАЛНА ПРИМЕНА У ОБРАЗОВАЊУ

Горан Геџин¹ Љубица Казин² Вишња Огњеновић³ Ивана Берковић⁴

Резиме: У овом раду је објашњена предност примене вишеслојности у развоју апликација, шта представља приказ елемената у master-detail односу, као и различити начини остваривања истог. Дат је и пример апликације која имплементира ова два концепта у развоју апликација.

Кључне речи: master-detail, веб, апликација, учење, ресторан

DEVELOPMENT OF MASTER-DETAIL WEB APPLICATION - POTENTIAL LEARNING TOOL

Abstract: This paper explains the benefits of the multi-layered approach of application development, defines what master-detail presentation is and what are different approaches of implementing it. Featured in this paper is an example application that incorporates both discussed concepts of application development.

Key words: master-detail, web, application, learning, restaurant

1. УВОД

Годинама уназад примена Интернета у свим сферама живота расте, па је и развој веб апликација у сталном порасту. Због све чешћег коришћења мањих уређаја попут мобилних телефона и таблета, који корисницима омогућају да користе веб апликације било кад и било где, постоји велика потреба за веб апликацијама које се прилагођавају уређајима на којима се извршавају, односно приказују. Многи наставни материјали су похрањени на разне едукативне сајтове, а ученици имају потребу да неке делове материјала детаљније сагледају на лак и прегледан начин.

Пошто се данас софтвер развија у деловима, и то често од стране више различитих фирми, изражена је вишеслојност апликација. То значи да се велики део кода налази у различитим библиотекама класа на које се позивају друге класе, а сваки део је задужен само за једну улогу.

У овом раду је представљен случај апликације која је урађена кроз вишеслојну архитектуру са посебним освртом на израду страница са master-detail односом елемената. Master-detail је кључан када је потребно истовремено приказати и неку целину и делове који је сачињавају.

Апликација је развијена у ASP.NET окружењу које омогућава брзу израду страница коришћењем унапред направљених контрола које се лако постављају на форму, а затим им се додаје функционалност писањем програмског кода у C# програмском језику.

У односу на спецификацију захтева урађена је апликација која може да се користи како у угоститељским школама, тако и у угоститељству, односно у ресторанима.

¹ Мсц, Технички факултет „Михајло Пупин“, Буре Ђаковића бб, Зрењанин, gorancecin@gmail.com

² Доц. др, Технички факултет „Михајло Пупин“, Буре Ђаковића бб, Зрењанин, ljubica.kazi@gmail.com

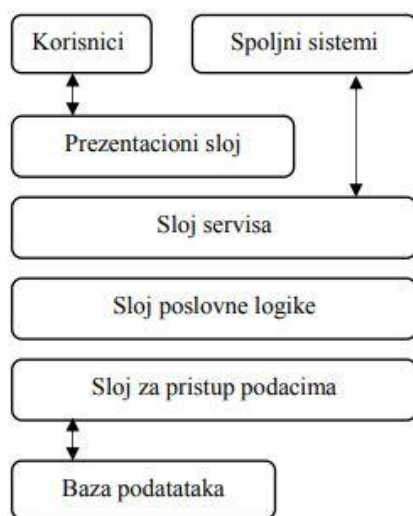
³ Доц. др, Технички факултет „Михајло Пупин“, Буре Ђаковића бб, Зрењанин, visnjao@tfzr.uns.ac.rs

⁴ Проф. др, Технички факултет „Михајло Пупин“, Буре Ђаковића бб, Зрењанин, berkovic@tfzr.uns.ac.rs

2. ПРИКАЗ ВИШЕСЛОЈНЕ АПЛИКАЦИЈЕ СА MASTER-DETAIL ТИПОМ ОДНОСА ЕЛЕМЕНАТА

Сваки софтвер се састоји од одређеног броја компоненти које имплементирају различите аспекте функционалности. У зависности од тога како су ове компоненте распоређене и повезане, разликују се једнослојне, двослојне, трослојне и вишеслојне апликације (Software Architecture, 2017).

Вишеслојна архитектура дели средишњи апликативни слој на нове слојеве од којих сваки има своју посебну функцију, односно намену. На Слици 1 је приказан класичан изглед слојева ASP.NET апликације, а пример у овом раду је урађен према истом обрасцу.



Слика 1 – Изглед архитектуре ASP.NET апликације (Иванковић, Лацмановић, 2016.)

Раздвајање апликације у слојеве има своје предности. Тиме се значајно повећава могућност поновног искоришћења постојећег кода, а значајно се олакшава процес одржавања, јер се неопходне измене раде само на једном конкретном делу апликације, без утицања на функционисање осталих делова.

2.1. Master-detail тип односа елемената апликације

Приликом развијања софтверских решења често се долази у ситуацију када је потребно да у исто време прикажемо неку целину и неке њене појединости. У тој ситуацији врло је zgodно искористити master-detail тип приказа. Такође, master-detail тип електронских форми се јавља и код уноса одговарајућих података у односу целина – део.

Master представља део корисничког интерфејса где у већини случајева имамо листу из које бирамо један елемент (целина), а detail је део у којем се приказују детаљнији подаци везани за изабрани master (Windows Apps Team, 2017).

Класичан пример за примену master-detail приказа јесте рачун у ставке које чине тај рачун или детаљи о особи која је изабрана из листе.

Једна од важних ствари за разматрање приликом коришћења ове технике је како ће контроле бити распоређене по форми странице. Данас се за прегледање и приступ апликацијама користе различити уређаји, па се стога и кориснички интерфејс мора прилагодити према циљном уређају. Сви ти уређаји имају различите величине екрана,

резолуције, чак и различите оријентације, тако да је немогуће имати само једну имплементацију.

За уређаје високе резолуције и са хоризонталном оријентацијом преферира се коришћење „side by side“ приказа, односно контроле које приказују податке из „detail“ ређају се одмах до „master“-а у једну страну. За уређаје мање резолуције и оне са вертикално оријентисаним екранима, углавном се користи приказ једног испод другог, или се користи замена страница. Тиме се приликом избора једног „master“-а прелази на нову страницу која приказује детаље.

За master-detail се може направити паралела са везом један према више (1-M) из релационих база података, јер се приликом уноса у базу података, подаци уписују у више табела (табела целина и табела ставке) применом више INSERT INTO наредби. При том је неопходно користити трансакцију која обухвата све SQL наредбе и извршава их пакетно: или све или ни једну. Читање података за изабрани master реализује се или из више различитих табела, или се учитава више редова из једне табеле.

3. ПРИМЕР АПЛИКАЦИЈЕ ЗА РЕСТОРАН СА МОГУЋНОШЋУ УЧЕЊА И МОДИФИКАЦИЈЕ РЕЦЕПАТА

Ресторан који поред менија има и детаљан приказ рецепта јела које нуди, је одлична понуда. Тиме се потесијалном посетиоцу олакшава избор јела, нарочито у случајевима алергије на неку храну или у случају болести када је веома битно колико дуго је јело пржено као и у другим случајевима. За посетиоце ресторана који су били задовољни одређеним јелом, а желели би да га сами спреме код куће, детаљан рецепт је свакако одлична лекција из кувања. Оно што је нарочити куриозитет који је урађен у овом примеру, је могућност логовања на систем и модификовање рецепта од стране посетиоца. Тиме посетилац јасно објашњава кувару шта хоће да се модификује у припреми одређеног јела које ће следећи пут да наручи.

3.1. Спецификација затева

Захтеви корисника приказани су у Табели 1 и Табели 2.

Табела 1 – Спецификација захтева за профил Кувар

Софтверска функција	Имплементација софтверске функције
Унос рецепта	Форма за унос са могућношћу додавања више састојака
Измена рецепта и брисање	Форма преко које ће се вршити измена и брисање рецепта
Преглед рецепта	Приказ листе рецепата и података за изабрани рецепт
Штампање рецепта	Страница прилагођена за штампање са изабраним рецептом

Табела 2 – Спецификација захтева за профил Посетилац

Софтверска функција	Имплементација софтверске функције
Преглед понуде јела (рецепата)	Форма за приказ листе рецепата по категорији
Штампа понуде јела (рецепата)	Страница са приказом табеле јела

3.2. Приказ структуре софтвера

У Табели 3 дат је приказ структуре софтвера.

Табела 3 – Приказ имплементираних делова развијене вишеслојне апликације

Главни слој	Подслој	Опис имплементације
Презентациони слој	Кориснички интерфејс	ASP.NET веб форме (Login.aspx, KuvarPregledRecepta.aspx)
	Класе презентационе логике	DLL библиотека класа (clsPosetilacPregledJela, clsKuvarIzmenaForma)
Сервисни слој	Веб сервис	Пројекат типа веб сервис - on-line библиотека класа којој се приступа ради удаљеног извршавања кода
	Класе за мапирање слојева	Нису имплементирани у примеру
Слој пословне логике	Радни токови	Нису имплементирани у примеру
	Пословни објекти	Нису имплементирани у примеру
	Пословне правила	DLL библиотека класа (clsIspitivanjeSastojaka)
Слој за рад са подацима	Рад са релационом базом података	DLL библиотека класа (clsSastojakDB, clsReceptDB)
	Рад са другим форматима података - XML, JSON	DLL библиотека класа – све DB класе раде са XML фајловима, као и класа веб сервиса

3.3. Опис примењене технологије

У изради апликације коришћен је Microsoft Visual Studio, .NET Framework и ASP.NET технологија. Конкретно коришћене су ASP.NET веб форме. Оне развијају функционалност – скриптови писани у C# језику који користе објекте и функције из других библиотека, од корисничког интерфејса који чине HTML и CSS (Penberthy, 2016).

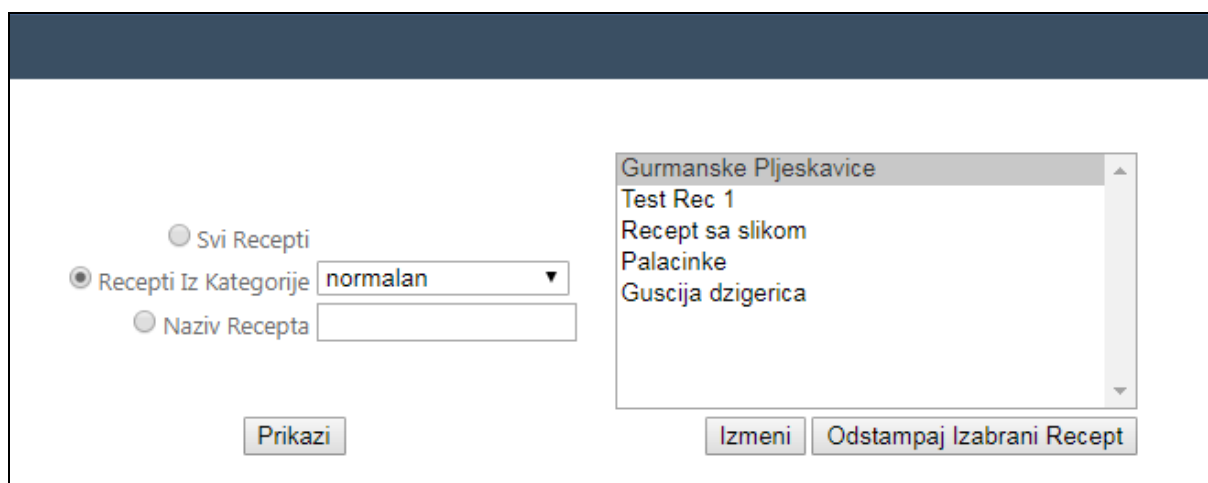
3.4. Кориснички интерфејс

На овом месту ће се приказати изглед неких карактеристичних форми. Форма за понуду јела је приказана на Слици 2:



Слика 2 – Изглед екрана са приказом свих јела из понуде

Након успешног пријављивања отвара се страница на којој се претражују и прегледају рецепти. Претраживање је могуће преко категорије рецепта или дела назива рецепта (Слика 3).



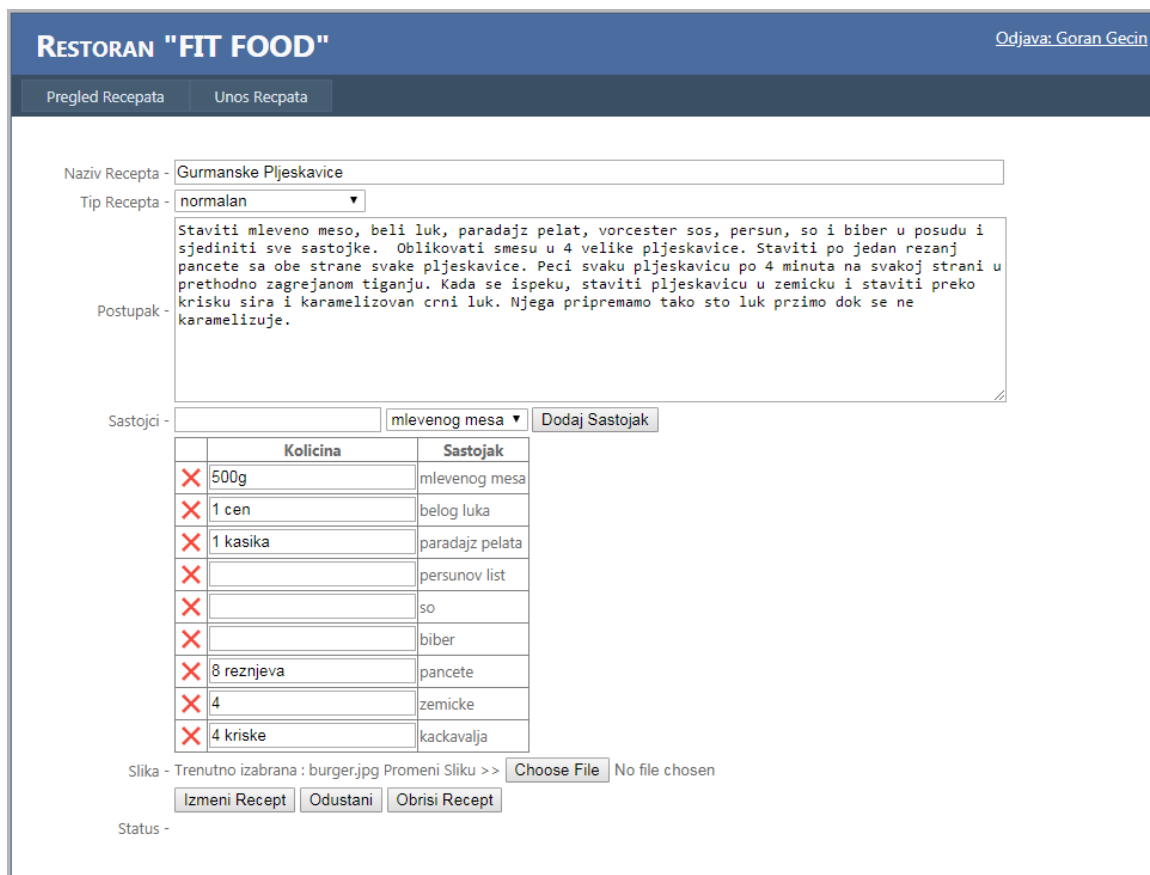
Слика 3 – Страница за преглед рецепата - претраживање по категорији рецепта

Када се изабере један рецепт из листе рецепата у доња два поља се појављује поступак за припрему тог рецепта и потребни састојци, а уколико постоји, учитава се и слика (Слика 4).



Слика 4 – Master – detail страница за приказ детаља изабраног јела

Кликом на дугме „Izmeni“ отвара се страница на којој се могу извршити измене података изабраног рецепта (Слика 5).



Слика 5 – Страница за измену и брисање изабраног рецепта

4. ЗАКЉУЧАК

У раду је показан начин пројектовања вишеслојне апликације са посебним освртом на израду страница са master-detail односом елемената. Допринос овог рада је приказ могућности master-detail типа електронских форми у оквиру учења рецепата. Више од тога, показан је начин индивидуалне модификације одређеног рецепта кроз master-detail екранске форме. Апликација је урађена као прототип који има велику практичну примену.

5. ЛИТЕРАТУРА

- [1] Penberthy W., Beginning ASP.NET 6: Web Forms and MVC, 2016
- [2] Software Architecture (2017), <http://www.softwaretestingmaterial.com/software-architecture/>
- [3] Windows Apps Team (2017), Master the Master-Detail Pattern, <https://blogs.windows.com/windowsdeveloper/2017/05/01/master-master-detail-pattern/>
- [4] Иванковић З., Лацмановић Д. (2016), Софтверско инжењерство 2, <http://www.tfzr.uns.ac.rs/Content/files/1/SE2%20Ivankovic%20Lacmanovic.pdf>