

UNIVERZA V LJUBLJANI  
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Dino Ivan Tot

IZDELAVA ASP .NET SPLETNE APLIKACIJE OD NAROČILA DO IZDELKA

DIPLOMSKO DELO UNIVERZITETNEGA ŠTUDIJA

Ljubljana, 2008

UNIVERZA V LJUBLJANI  
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Dino Ivan Tot

IZDELAVA ASP .NET SPLETNE APLIKACIJE OD NAROČILA DO IZDELKA

DIPLOMSKO DELO UNIVERZITETNEGA ŠTUDIJA

Mentor:

doc. dr. Marko Bajec, univ. dipl. ing.

Ljubljana, 2008

# Zahvala

Zahvaljujem se mentorju doc. dr. Marku Bajcu, univ. dipl. ing. za vodenje in prijazno pomoč pri izdelavi moje diplomske naloge.

Zahvaljujem se tudi vsem profesorjem Fakultete za računalništvo in informatiko, ki so mi omogočili pridobitev širokega spektra računalniškega znanja, prijazno doživetje študija in pripomogli pri premagovanju jezikovne razlike na začetku študija.

Posebej se zahvaljujem staršem, da so mi omogočili študij, ki sem si ga želel.

## Kazalo

Kazalo slik .....	7
Kazalo tabel .....	8
Seznam uporabljenih kratic.....	10
Povzetek .....	11
Abstract .....	12
1 Uvod .....	13
2 Opredelitev problema.....	14
2.1 Lista zahtev, ki jih kot izvajalec želimo izpolniti .....	14
2.2 Tehnološke omejitve naročnika .....	14
2.3 Vsebinske zahteve naročnika .....	15
3 Tehnološka platforma .....	18
3.1 .NET ogrodje 3.5 .....	18
3.2 ASP .NET .....	18
3.3 AJAX .....	18
3.4 ASP .NET AJAX.....	19
3.5 IIS - Internet Information Services.....	19
3.6 Microsoft Visual Studio .....	19
3.7 Microsoft SQL Server .....	20
3.8 Microsoft Project .....	20
3.9 CSS.....	20
4 Ocena časa potrebnega za izvedbo .....	21
5 Izdelava dokumenta korakov prevzemnih testov .....	23
6 Projektno vodenje in izdelava projektnega plana .....	26
6.1 Sistem projektnega vodenja.....	26
6.2 Izdelava projektnega plana z uporabo orodja Microsoft Project.....	27
7 Planiranje arhitekture aplikacije.....	30
8 Izdelava dizajna internetnih strani aplikacije .....	32
9 Modeli primerov uporabe .....	34
9.1 Iskanje namestitve .....	34
9.2 Pregled apartmaja .....	35

9.3	Registracija .....	36
9.4	Prijava.....	37
9.5	Nalaganje slik.....	38
9.6	Vnos podatkov o apartmaju .....	39
9.7	Vnos kontaktnih podatkov lastnika .....	39
9.8	Izbira dodatne opreme .....	40
9.9	Vzdrževanje šifranta jezikov.....	41
9.10	Vzdrževanje šifranta dodatne opreme .....	43
9.11	Vzdrževanje šifranta mest.....	43
9.12	Vzdrževanje šifranta regij.....	44
9.13	Vzdrževanje šifranta tipov apartmaja.....	45
9.14	Vzdrževanje šifranta prevodov.....	45
10	Izdelava aplikacije .....	47
10.1	Sloj podatkov .....	47
10.1.1	Konceptualni podatkovni model .....	47
10.1.2	Logični podatkovni model .....	51
10.1.3	Fizični podatkovni model .....	52
10.2	Sloj dostopa do podatkov .....	54
10.3	Sloj poslovne logike .....	55
10.3.1	Razred »Appartment«.....	56
10.3.2	Razred "AppType" .....	56
10.3.3	Razred "Language" .....	56
10.3.4	Razred "Owner" .....	57
10.3.5	Razred "PropertyInformation" .....	57
10.3.6	Razred "Region" .....	57
10.3.7	Razred "Town" .....	58
10.4	Sloj uporabniškega vmesnika .....	59
10.4.1	Modul za iskanje apartmajev .....	59
10.4.2	Modul za prikaz seznama rezultatov iskanja .....	59
10.4.3	Modul za registracijo uporabnika.....	60
10.4.4	Modul za prijavo uporabnikov v sistem.....	61
10.4.5	Modul za prikaz tri najboljiskana mesta .....	61

10.4.6	Modul za prikaz galerije slik .....	62
10.4.7	Modul za prikaz tipa apartmaja.....	62
10.4.8	Modul za prikaz števila postelj .....	63
10.4.9	Modul za prikaz opreme apartmaja .....	63
10.4.10	Modul za prikaz kontaktnih podatkov lastnika.....	64
10.4.11	Modul za prikaz cen po sezonah.....	64
10.4.12	Modul za direkten kontakt z lasnikom .....	65
10.4.13	Modul za nalaganje slik v galerijo .....	66
10.4.14	Modul za vpis podatkov o apartmaju.....	67
10.4.15	Modul za vnos opreme apartmaja .....	68
10.4.16	Modul za vpis kontaktnih podatkov lastnika .....	69
11	Omejevanje dostopa do strani za registrirane uporabnike .....	70
12	Lokalizacija strani.....	71
12.1	Lokalizacija statičnih elementov ASP .NET internetnih strani .....	71
12.2	Lokalizacija dinamičnih elementov.....	72
13	Postavitev aplikacije na produkcijski strežnik .....	73
14	Sklepne ugotovitve .....	74
	Priloge .....	76
1.	SQL ukazi za kreacijo fizične podatkovne baze.....	76
2.	XSD shema datotek za lokalizacijo .....	79
3.	Glavni razred .....	80
4.	Realizacija razreda "Town" sloja poslovne logike .....	82
5.	Uporabniški vmesnik aplikacije.....	84
	Viri (Literatura) .....	98

## Kazalo slik

Slika 6-1 – Pogled na projekt čez trikotnik čas, strošek, obseg .....	27
Slika 6-2 - Projektne aktivnosti grupirane v projektne faze .....	28
Slika 6-3 – Gantov diagram projekta.....	29
Slika 7-1 – Prikaz večslojne arhitekture .....	30
Slika 8-1 – Izdelava grafike strani v orodju za urejanje grafike .....	32
Slika 8-2 – Glavna stran aplikacije .....	33
Slika 8-3 – Navigacijski sistem formatiran na način 1.....	33
Slika 8-4 - Navigacijski sistem formatiran na način 2.....	33
Slika 9-1 - Model primerov uporabe "Iskanje namestitve".....	34
Slika 9-2 - Model primerov uporabe "Registracija in prijava" .....	36
Slika 9-3 - Model primerov uporabe "Vnos in vzdrževanje podatkov o apartmaju" .....	38
Slika 9-4 - Model primerov uporabe "Administracija".....	42
Slika 10-1 – Konceptualni podatkovni model aplikacije »Direktno od naročnika«.....	47
Slika 10-2 – Logični podatkovni model aplikacije »Direktno od naročnika« .....	51
Slika 10-3 – Fizični podatkovni model aplikacije »Direktno od naročnika« .....	52
Slika 10-4 – Fizični podatkovni model modula za registracijo in prijavo uporabnikov.....	53
Slika 10-5 – Prikaz ADO .NET podatkovnega sloja .....	54
Slika 10-6 – Razredni diagram sloja poslovne logike .....	55
Slika 10-7 – Uporabniški vmesnik modula za iskanje apartmajev.....	59
Slika 10-8 – Uporabniški vmesnik modula za prikaz seznama rezultatov iskanja .....	60
Slika 10-9 - Uporabniški vmesnik modula za registracijo uporabnika .....	60
Slika 10-10 - Uporabniški vmesnik modula za prijavo prikazan v hrvaščini .....	61
Slika 10-11 - Uporabniški vmesnik modula za prikaz tri najbolj iskanih mest .....	61
Slika 10-12 - Privzeta slika .....	61
Slika 10-13 - Uporabniški vmesnik modula galerija .....	62
Slika 10-14 – Uporabniški vmesnik modula za prikaz tipa apartmaja .....	62
Slika 10-15 - Uporabniški vmesnik modula za prikaz števila postelj .....	63
Slika 10-16 - Uporabniški vmesnik modula za prikaz opreme apartmaja.....	63
Slika 10-17 - Uporabniški vmesnik modula za prikaz kontaktnih podatkov lastnika .....	64
Slika 10-18 - Uporabniški vmesnik modula za prikaz cen .....	64
Slika 10-19 - Uporabniški vmesnik modula za direkten kontakt s lastnikom.....	65
Slika 10-20 - Uporabniški vmesnik modula za nalaganje slik .....	66
Slika 10-21 - Uporabniški vmesnik modula za vpis podatkov o apartmaju .....	67
Slika 10-22 - Uporabniški vmesnik modula za vnos opreme.....	68
Slika 10-23 - Uporabniški vmesnik modula za vpis kontaktnih podatkov lastnika.....	69
Slika 12-1 – Globalni viri .....	72
Slika 12-2 - Forma za kreiranje in posodabljanje prevodov Regij .....	72
Slika 13-1 – Konfiguracija aplikacije .....	73

Slika 0-1 - Internetna stran "Iskanje namestitve" .....	84
Slika 0-2 - Internetna stran "Registracija vašega apartmaja" .....	85
Slika 0-3 - Internetna stran "O projektu" .....	86
Slika 0-4 - Internetna stran "Pregled rezultatov" .....	87
Slika 0-5 - Internetna stran "Pregled apartmaja" .....	88
Slika 0-6 - Internetna stran "Prijava" .....	89
Slika 0-7 - Internetna stran "Spreminjanje in vnos podatkov o apartmaju" .....	90
Slika 0-8 - Internetna stran "Registracija" .....	91
Slika 0-9 - Internetna stran "Administracija regij" .....	92
Slika 0-10 - Internetna stran "Administracija mest" .....	93
Slika 0-11 - Internetna stran "Administracija jezikov" .....	94
Slika 0-12 - Internetna stran "Administracija dodatkov" .....	95
Slika 0-13 - Internetna stran "Administracija tipov" .....	96
Slika 0-14 - Internetna stran "Administracija prevodov" .....	97

## Kazalo tabel

Tabela 1 - Ocena časa izvedbe.....	21
Tabela 2 - Plan prevzemnih testov .....	25
Tabela 3 - Seznam relacij entitete »Appartmant«.....	48
Tabela 4 - Seznam atributov entitete »Appartmant«.....	48
Tabela 5 - Seznam relacij entitete »Appartment_type«.....	48
Tabela 6 - Seznam atributov entitete »Appartment_type«.....	48
Tabela 7 - Seznam relacij entitete »Extras« .....	48
Tabela 8 - Seznam atributov entitete »Extras« .....	48
Tabela 9 - Seznam relacij entitete »Language«.....	49
Tabela 10 - Seznam atributov entitete »Language«.....	49
Tabela 11 - Seznam relacij entitete »News« .....	49
Tabela 12 - Seznam atributov entitete »News« .....	49
Tabela 13 - Seznam relacij entitete »Owner« .....	49
Tabela 14 - Seznam atributov entitete »Owner« .....	49
Tabela 15 - Seznam relacij entitete »Region« .....	50
Tabela 16 - Seznam atributov entitete »Region« .....	50
Tabela 17 - Seznam relacij entitete »Town« .....	50
Tabela 18 - Seznam atributov entitete »Town« .....	50
Tabela 19 - Lastnosti razreda Appartment.....	56
Tabela 20 - Metode razreda apartment.....	56
Tabela 21 - Lastnosti razreda AppType .....	56
Tabela 22 - Metode razreda AppType .....	56



Tabela 23 - Lastnosti razreda Language .....	56
Tabela 24 - Metode razreda Language .....	57
Tabela 25 - Lastnosti razreda Owner .....	57
Tabela 26 - Metode razreda Owner .....	57
Tabela 27 - Lastnosti razreda PropertyInformation .....	57
Tabela 28 - Metode razreda PropertyInformation .....	57
Tabela 29 - Lastnosti razreda Region .....	58
Tabela 30 - Metode razreda Region .....	58
Tabela 31 - Lastnosti razreda Town .....	58
Tabela 32 - Metode razreda Town .....	58

## Seznam uporabljenih kratic

CLR - Common Language Runtime, predstavlja virtualni stroj v .NET ogrodju

ASP - Active Server Pages, tehnologija za avtomatično ustvarjanje spletnih strani na strežniku

SQL - Structured Query Language, jezik za manipulacijo z podatki

.NET – Microsoft .NET ogrodje

JPEG – format bitne slike

PNG - format bitne slike

GIF - format bitne slike

DHTML - Dynamic Hyper Text Markup Language, skupina tehnologij za kreiranje interaktivnih spletnih strani

RDBMS - Relational Database Management System, sistem za upravljanje z relacijskimi podatkovnimi bazami

## Povzetek

V diplomskem delu je opisan proces razvoja spletne aplikacije v tehnologiji ASP .NET od naročila do izdelka na primeru aplikacije za podporo spletnem oglaševanju apartmajev privatnih ponudnikov.

V poglavjih 3 do 7 so opisani koraki vzpostavitve projekta. Delo se koncentrira na določanje projektnih aktivnosti, časa izvedbe posameznih modulov in izdelavo dokumenta prevzemnih testov. Opisan je način projektnega vodenja in izdelave projektnega plana.

Poglavja 8 do 11 govorijo o tehnični izvedbi projekta. Izvedba se začne z opisom izbire arhitekture rešitve. Temu sledijo opisi primerov uporabe in izdelava dizajna internetnih strani aplikacije. Poglavje 9 govori o konkretni izdelavi aplikacije. Izdelava se temelji na večslojni arhitekturi. V ta namen je opisan vsak sloj arhitekture: sloj podatkov, sloj dostopa do podatkov, sloj poslovne logike in sloj uporabniškega vmesnika.

Preostala poglavja govorijo o končnih aktivnostih na projektu kot so lokalizacija strani in sistem za omejevanje dostopa do strani za neregistrirane uporabnike. Kot zadnji korak projekta pred podpisom prevzemnih testov je opisana postavitve aplikacije na produkcijski strežnik.

**KLJUČNE BESEDE:** *ASP .NET, vzpostavitev projekta, večslojna arhitektura, lokalizacija*

## Abstract

This diploma work describes the process of developing Web applications in ASP . NET technology from the contract to the product. Process description is based on development of application for online advertising support of apartments owned by private individuals.

Chapters 3 to 7 describe the steps of setting up the project. Work is concentrated on the determination of project activities, the time for implementation individual modules and writing acceptance test plan. Chapters also describe the way of project management and construction of the project plan.

Chapter 8 to 11 speak on the technical execution of the project. Implementation begins with a description of the choice of architectural solutions. This is followed by a use case descriptions and manufacture of Internet sites design. Chapter 9 talks about the manufacture of core application. Manufacture is based on Multi-Tier-architecture. For this purpose each layer of architecture is described: the data layer, data access layer, business logic layer and user interface.

The remaining chapters are about the final activity on the project such as the localization and a system for restricting access to the site for unregistered users. As the final step in the project before signing the acceptance tests is described process of application deployment to a production server.

**KEY WORDS:** *ASP .NET, Project Set-up, Multi-Tier Architecture, Localization*

## 1 Uvod

Ideja o potrebi opisa izdelave spletne ASP .NET aplikacije je prišla iz problemov s katerim sem se srečal v praksi. Velikokrat se je zgodilo da je na projektu zapravljeno preveč časa v določenih fazah projekta. Zato sem se odločil da v tem delu prikažem vse smiselne korake projekta izdelava ASP .NET spletne aplikacije od naročila do izdelka.

Moj cilj je prikazati kako z uporabo mehkih arhitekturnih rešitvah in učinkovito postavitvijo projekta v začetni fazi, lahko projekt izpeljemo pravočasno in brez dodatnih stroškov, brez vpliva na kvaliteto in zmogljivost.

Za izvršitev te naloge bom uporabil vsa razpoložljiva orodja za pomoč izvedbi projekta. Korake projekta bom prikazal na primeru aplikacije za podporo spletnem oglaševanju apartmajev privatnih ponudnikov.

## 2 Opredelitev problema

Potrebno je izdelati spletno aplikacijo za pregled ponudbe apartmajev privatnih ponudnik. Aplikacijo je potrebno izdelati v roku treh mesecev. Naročnik ima proračun v okviru cene 90 inženirskih dni.

### 2.1 Lista zahtev, ki jih kot izvajalec želimo izpolniti

1. Uporabniški vmesnik je zelo pomemben. To je tista prva stvar, ki jo uporabnik opazi preden začne uporabljati funkcije strani. S tem ne želim reči da je pomembno narediti samo lepo grafiko. Potrebno je organizirati informacije na strani tako da so dobro organizirane in lahko dosegljive.
2. Nekaj pozornosti je potrebno posvetiti kompatibilnosti pregleda strani v internet pregledovalnikih različnih proizvajalcev. Ponavadi je dobro podpirati vsaj najnovejše verzije pregledovalnikov Microsoft Internet Explorer in Mozilla Firefox.
3. Uspešne strani so uspešne zaradi števila uporabnikov, ki jih redno obiskuje. Zato je potrebno narediti sistem za registracijo in avtentikacijo uporabnikov
4. Stran rabi konstantno osveževanje vsebine. Edino tako lahko ostane interesantna uporabnikom. Če vsebina ostane dlje časa nespremenjena, uporabniki zgubijo interes in se ne vračajo na stran. V ta namen bo izdelan sistem za upravljanje z vsebino strani (CMS). Sistem bo omogočil osveževanje z strani tehničnih nestrokovnjakov.
5. Enkrat ko bo stran postavljena in se redno osvežuje, bo potrebno na nek način obvestiti uporabnike o spremembah. To je zato, ker večina uporabnikov ne spremlja spremembe na strani vsak dan. Registrirani uporabniki so pri registraciji vpisali svojo elektronsko pošto tako da ta podatek lahko uporabimo v namen obveščanja.
6. V testni fazi aplikacije bo potrebno omogočiti pošiljanje napak aplikacije na elektronsko pošto. Potrebno bo tudi zagotoviti zapisovanje napak v tekstovno datoteko.
7. V drugi fazi projekta bo potrebno planirati tudi uporabniški forum. To je eden izmed najlažjih načinov kako zadržati uporabnike. Večina radi bere komentarje drugih ljudi.

### 2.2 Tehnološke omejitve naročnika

Naročnik že ima nameščen internetni strežnik na katerem je operacijski sistem Windows 2003. Naročnikov administrator, ki bo skrbel za novo internetno aplikacijo je usposobljen za administracijo Microsoft IIS aplikacijskega strežnika. Pri naročniku trenutno ni nameščena nobena podatkovna baza tako da tudi nima plačane licence za nobeno. Naš predlog

naročniku je uporaba MS SQL Express podatkovne baze. Prednost te izbire je hiter preklon na plačljivo verzijo, ki je arhitekturno povsem enaka brezplačni.

## 2.3 Vsebinske zahteve naročnika

1. Sistem za registracijo in prijavo uporabnikov
  - a. Potrebno je omogočiti registracijo uporabnikov v sistem. Obvezni podatki pri registraciji so elektronski naslov, ime in priimek.
  - b. Aplikacija mora podpirati različne stopnje pravic. Prijava mora biti enostavna in omogočena z vseh strani aplikacije.
  
2. Jezikovno podporo za več jezikov
  - a. Stran bo namenjena uporabnikom različnih govorečih področij. Torej bo potrebno izdelati podporo za izbiro jezika. V prvi fazi projekta bo izbira omejena na tri jezike: Slovenščino, Hrvaščino in Angleščino.
  - b. Aplikacija mora omogočiti dodajanje novih jezikov poleg tri osnovna in administracijo prevodov.
  
3. Modul za iskanje apartmajev, ki omogoča:
  - a. Iskanje se mora izvajati po geografskem kriteriju. Aplikacija ponudi uporabniku na izbiro geografske regije in po na podlagi te izbire tudi pripadajoča mesta.
  - b. Uporabnik lahko omeji iskanje vnosom števila oseb ter najvišje in najnižje cene za dan uporabe apartmaja
  
4. Modul za prikaz seznama iskanih apartmajev, ki omogoča:
  - a. Na podlagi kriterij, ki mu ga pošlje modul za iskanje prikaže seznam vseh apartmajev, ki zadoščajo tem kriterijem
  - b. Mora omogočiti prikaz dolgega seznama na več strani
  - c. Omogočiti je potrebno konfiguracijo števila prikazanih apartmajev na eni strani
  
5. Modul za prikaz posameznega apartmaja, ki omogoča:
  - a. Prikaz do 6 malih slik v enem od naslednjih formatov: jpg, png, gif.
  - b. Prikaz kontaktnih podatkov lastnika apartmaja
  - c. Prikaz tipa nastanitve
  - d. Prikaz števila ležajev in pomožnih ležajev
  - e. Prikaz seznama dodatne opreme, ki je apartma vsebuje
  - f. Prikaz cen za predsezono, sezono in posezono

- g. Prikaz forme za pošiljanje vprašanj direktno na lastnikovo elektronsko pošto. Vsebovati mora polja telefon, elektronski naslov, datum prihoda, datum odhoda, število odraslih ljudi, število otrok, polje za opis vprašanja.
6. Modul za vnos in administracijo apartmaja, ki omogoča:
    - a. Pošiljanje slik na strežnik v formatih jpg, gif in png poljubne velikosti. Poskrbeti mora za ustrezno zmanjšanje slike in shranjevanje slike.
    - b. Vnos kontaktnih podatkov lastnika apartmaja
    - c. Vnos tipa nastanitve
    - d. Vnos števila ležajev in pomožnih ležajev
    - e. Izbiro seznama dodatne opreme, ki je apartma vsebuje
    - f. Vnos cen za predsezono, sezono in posezono
  7. Modul za prikaz tri najbolj iskana mesta, ki omogoča
    - a. Povečevanje številke popularnosti po vsakem iskanju mestu, ki je bilo izbrano
    - b. Slikovni prikaz treh mestih, ki imajo najvišje številke popularnosti
  8. Module za administracijo šifrantov
    - a. Modul za administracijo seznama jezikov
    - b. Modul za administracijo dodatne opreme apartmajev in administracijo njihovih prevodov
    - c. Modul za administracijo tipov apartmaja
    - d. Modul za administracijo regij in administracijo njihovih prevodov
    - e. Modul za administracijo mest in administracijo njihovih prevodov
  9. ASP.NET internet strani
    - a. »Glavna stran«: sestavljena mora biti iz glave strani, na kateri so povezave do strani za registracijo in prijavo, gumbi za spremembo jezika in kazalo strani. V vsebini glavne strani bodo prikazane vse ostale strani
    - b. »Iskanje namestitev«: Prikazuje modul za iskanje, modul za prikaz tri najbolj iskana mesta in prikazuje statične podatke o državi
    - c. »O Projektu«: Prikazuje kontaktne podatke podjetja. Vsebuje modul za pošiljanje sporočila na naslov podjetja.
    - d. »Registracija apartmaja«: Prikazuje podatke o ceni oglaševanja apartmaja, pogoje oglaševanja in vsebuje opis postopka registracije.
    - e. »Registracija«: Vsebuje modul za registracijo uporabnika.
    - f. »Prijava«: Vsebuje modul za prijavo.
    - g. »Prikaz rezultatov iskanja«: Vsebuje modul za prikaz seznama apartmajev.
    - h. »Apartma«: Vsebuje modul za prikaz podatkov o posameznem apartmaju
    - i. »Administracija apartmaja«: Vsebuje modul za spreminjanje in dodajanje podatkov o apartmaju. Dostopen je samo prijavljenim uporabnikom.



- j. »Administracija regij«: Omogoča vnašanje in spreminjanje seznama regij.
- k. »Administracija mest«: Omogoča vnašanje in spreminjanje seznama mest.
- l. »Administracija jezikov«: Omogoča vnašanje in spreminjanje seznama jezikov.
- m. »Administracija dodatkov apartmaja«: Omogoča vnašanje in spreminjanje seznama dodatkov.
- n. »Administracija prevodov: Omogoča vnašanje in spreminjanje prevodov vrednosti vsebine šifrantov regij, jezikov in dodatkov.

### 3 Tehnološka platforma

Kot razvojno okolje bo uporabljeno orodje podjetja Microsoft, Visual Studio Team System 2008. Aplikacijski strežnik bo Microsoft IIS. Za podatkovno bazo bo uporabljen MS SQL Server 2005. Microsoft ASP .NET 3.5, jezik C# bo uporabljen za programiranje spletne aplikacije. Za asinhrono klice pa tehnologija Microsoft AJAX, verzija 1.0.

#### 3.1 .NET ogrodje 3.5

[18] Microsoft .NET ogrodje je tehnologija, ki vsebuje veliko število že napisanih rešitev za znane probleme v programiranju. Sestavljajo jo že napisane knjižnice in CLR<sup>1</sup> virtualni stroj. Virtualni stroj upravlja z vsemi programi napisanimi v enem od jezikov podprtih z .NET ogrodjem. Daje tudi podporo pomembnim programskim storitvam kot so varnost, upravljanje s spominom, upravljanje z napakami itn.

#### 3.2 ASP .NET

[9] Microsoft ASP.NET je brezplačna tehnologija, ki omogoča programerjem ustvarjanje dinamičnih spletnih aplikacij.

ASP .NET programerji lahko uporabijo za izgradnjo dinamičnih spletnih strani, spletnih aplikacij in spletnih storitvah. Najprej je bila sproščena v januarju 2002 za verzijo 1.0 .NET ogrodja. Je naslednik Microsoft ASP<sup>2</sup> tehnologije. ASP .NET je zgrajen na CLR-ju, ki omogoča programerjem pisanje ASP.NET kode z uporabo katerega koli .NET jezika.

#### 3.3 AJAX

[8] AJAX je skupina medsebojno povezanih spletnih tehnik, uporabljenih za ustvarjanje interaktivnih spletnih aplikacij. Z AJAX-om lahko prikličete podatke iz strežnika asinhrono v ozadju, brez motenj na zaslonu in spremembe obnašanja obstoječe strani.

---

<sup>1</sup> Common Language Runtime – predstavlja virtualni stroj v .NET ogrodju

<sup>2</sup> Active Server Pages – tehnologija za avtomatično ustvarjanje spletnih strani na strežniku

### 3.4 ASP.NET AJAX

[11] ASP.NET AJAX je niz razširitvah Microsoft ASP.NET, ki so jih razvili za izvajanje AJAX funkcionalnost.

Komponente omogočajo ustvarjanje spletnih aplikacij v ASP.NET 2.0 na način da lahko posodobite podatke na spletni strani brez popolnega pošiljanja strani nazaj strežniku. Ključne tehnologije, ki omogoča, da se ta funkcija izvede je »XMLHttpRequest« predmet, skupaj s JavaScript-om in DHTML<sup>3</sup>.

ASP.NET AJAX je bil narejen kot samostojen dodatek za ASP.NET januarja 2007 po dolgem obdobju beta testiranja. Kasneje je bil vključen v različico 3.5 NET Framework-a, ki je bila sproščena ob sprostitvi Visual Studio 2008 v novembru 2007.

### 3.5 IIS - Internet Information Services

[13] Internet Information Services (IIS) - nekdanji imenovani Internet Information Server - je niz internetnih storitev za strežnike, ki jih ponuja Microsoft za uporabo v operacijskem sistemu Microsoft Windows. Je drugi najbolj priljubljeni spletni strežnik na svetu.

### 3.6 Microsoft Visual Studio

[17] Microsoft Visual Studio je glavno razvojno okolje, ki ga ponuja Microsoft. Lahko se uporablja za razvoj konzolnih in grafičnih aplikacij, spletnih strani, spletnih aplikacij in spletnih storitev za platforme: Microsoft Windows, Windows Mobile, Windows CE, .NET Framework, .NET Compact Framework in MicrosoftSilverlight.

Visual Studio ima vgrajeno podporo za jezike C / C (preko Visual C), VB.NET (preko Visual Basic. NET), in C # (z Visual C #). Podpora za druge jezike, kot so krom, F #, Python, Ruby in drugi je na voljo preko dodatkov, ki jih je treba namestiti ločeno. Prav tako podpira XML / XSLT, HTML / XHTML, JavaScript in CSS. Specifične različice jezikov za Visual Studio , ki tudi obstajajo, določajo bolj omejene jezikovne storitve za uporabnika. Ti posamezni paketi so: Microsoft Visual Basic, Visual J #, Visual C # in Visual C.

---

<sup>3</sup> Dinamic HyperText Markup Language – skupina tehnologij za kreiranje interaktivnih spletnih strani

### 3.7 Microsoft SQL Server

[16] Microsoft SQL Server, je sistem za upravljanje z relacijskim podatkovnim bazami (RDBMS<sup>4</sup>), ki ga proizvaja Microsoft. Njegova glavna jezika za poizvedovanje sta MS-SQL in T-SQL.

### 3.8 Microsoft Project

[14] Microsoft Project je produkt, ki je namenjen za pomoč projektnim vodjem pri razvoju načrtov finančnih sredstev, za sledenje napredku projekta, za upravljanje proračuna in analiziranje delovne obremenitve. Prva različica Microsoft Project-a je bila v DOS-a v letu 1984 s strani družbe, ki dela za Microsoft. Microsoft je kupil vse pravice za programsko opremo v letu 1985 in sprostil različico 2. Različica 3 za DOS je bila izdana leta 1986. Prva različica za operacijski sistem Windows je izšla leta 1990, in je bila označena kot različica 1 za operacijski sistem Windows. Zanimiva stvar, ki jo je treba omeniti, da je prva različica za DOS prvič uvedla koncept linija, ki povezujejo aktivnosti v Gantovem diagramu.

### 3.9 CSS

[12] CSS je jezik namenjen ločevanju vsebine dokumenta od načina predstavitve te vsebine. Ta ločitev lahko izboljša dostopnost vsebine, zagotovi večjo prožnost in nadzor ter zmanjša zapletenost in ponavljanje v strukturi vsebine.

---

<sup>4</sup> Relational Database Management System

## 4 Ocena časa potrebnega za izvedbo

Vsebinske zahteve naročnika prepisemo kot seznam aktivnosti in jim določimo čase potrebne za izvedbo. Opredelimo čas potreben za izdelavo specifikacij, razvoj in testiranje posameznega modula. Posameznem času dodamo čas garancija, ki je 10% vrednost razvojnega časa in je mišljen kot čas za popravilo napak v garancijskem roku.

<b>NASLOV MODULA</b>	<b>SPECIFIKACIJE</b>	<b>RAZVOJ</b>	<b>TEST</b>	<b>GARANCIJA</b>	<b>SKUPAJ</b>
Analiza uporabniških zahtev	3	0	0	0	3
Analiza okolja	1	0	0.5	0	1.5
Izdelava podatkovnega modela	2	2	1	0.4	5.4
Vzpostavitev razvojnega okolja	0	2	1	0	3
Modul za več jezikovno podporo	0.5	2	1.5	0.4	4.4
Modul za iskanje apartmajev	0.2	1	1	0.2	2.4
Modul za prikaz seznama iskanih apartmajev	0.2	2	2	0.4	4.6
Modul za prikaz posameznega apartmaja	0.5	2	2	0.4	4.9
Modul za vnos in administracijo apartmajev	0.1	2.5	2	0.5	5.1
Modul za registracijo in prijavo uporabnikov	0.2	5	3	1	9.2
Modul za prikaz tri najbolj iskana mesta	0.1	1	1	0.2	2.3
Modul za administracijo seznama jezikov	0	0.5	0.5	0.1	1.1
Modul za administracijo dodatne opreme	0	0.5	0.5	0.1	1.1
Modul za administracijo regij	0	0.5	0.5	0.1	1.1
Modul za administracijo mest	0	0.5	0.5	0.1	1.1
Module za administracijo prevodov šifrantov	0	0.5	0.5	0.1	1.1
CSS oblikovanje in optimizacija slik	1	3	0.3	0.6	4.9
ASP.NET glavna stran	0.2	0.5	0.1	0.1	0.9
ASP.NET stran "Iskanje namestitve"	0.2	0.5	0.1	0.1	0.9
ASP.NET stran "Registracija apartmaja"	0.2	0.3	0.1	0.06	0.66
ASP.NET stran "O Projektu"	0.2	0.2	0.1	0.04	0.54
ASP.NET stran "Prijava"	0	0.2	0.1	0.04	0.34
ASP.NET stran "Registracija"	0	0.2	0.1	0.04	0.34
ASP.NET stran "Seznam apartmajev"	0.2	0.2	0.1	0.04	0.54
ASP.NET stran "Pregled apartmaja"	0.2	0.2	0.1	0.04	0.54
ASP.NET stran "Vnos in urejanje apartmaja"	0	0.2	0.2	0.04	0.44
ASP.NET strani administracija šifrantov	0	1	0	0.2	1.2
Vzpostavitev testnega okolja pri naročniku	0.5	1	1	0.2	2.7
Prevzemni testi	1	0	1	0	2
Namestitev v produkcijsko okolje	0	1	3	0.2	4.2
Izdelava namestitvene dokumentacije	0.5	0	0	0	0.5
Uporabniški priročnik	1	0	0	0	1
Predstavitve pred uporabniki	0.5	0.5	0	0.1	1.1
<b>VSE SKUPAJ (ED)</b>	<b>13.7</b>	<b>31.5</b>	<b>23.9</b>	<b>5.9</b>	<b>75</b>

Tabela 1 - Ocena časa izvedbe

Ta seznam je podlaga za ceno produkta in podpis pogodbe. Zato je potrebno, da dokument napiše oseba, ki ima izkušnje in da si pri pisanju pomaga s razvijalci, ki bodo dejansko delali na posameznem modulu.

## 5 Izdelava dokumenta korakov prevzemnih testov

V dokumentu določimo, na podlagi dokumenta analize zahtev, vse funkcionalne zahteve, ki jih aplikacij mora imeti za ceno določeno v pogodbi. Dokument bo kontrolni seznam na prevzemnem testu.

Dokument je zelo pomemben zaradi odstopanja v željah naročnika tokom projekta. Pogodbo sestavimo tako da se vsa večja odstopanja v željah obravnavajo kot spremembe in se dodatno obračunajo.

MODUL ali STRAN	FUNKCIONALNA ZAHTEVA	OPRAVLJENO	PODPIS
Glavna stran	Glava strani vsebuje povezave do strani za registracijo in do strani za prijavo.	DA / NE	
	Vsebuje gumb za spremembo jezika.	DA / NE	
	Vsebuje kazalo strani.	DA / NE	
Iskanje namestitve	Prikazuje modul za iskanje	DA / NE	
	Prikazuje modul za prikaz tri najbolj iskana mesta	DA / NE	
	Prikazuje statične podatke o državi	DA / NE	
O projektu	Prikazuje kontaktne podatke podjetja	DA / NE	
	Vsebuje modul za pošiljanje sporočila na naslov podjetja	DA / NE	
Registracija apartmaja	Prikazuje podatke o ceni oglaševanja apartmaja,	DA / NE	
	Vsebuje opis postopka registracije	DA / NE	
Registracija	Vsebuje modul za registracijo uporabnika	DA / NE	
Prijava	Vsebuje modul za prijavo	DA / NE	
Prikaz rezultatov iskanja	Vsebuje modul za prikaz seznama apartmajev	DA / NE	
Apartma	Vsebuje modul za prikaz podatkov o posameznem apartmaju	DA / NE	
Administracija apartmaja	Vsebuje modul za spreminjanje in dodajanje podatkov o apartmaju	DA / NE	
	Dostopna je samo prijavljenim uporabnikom	DA / NE	

Administracija regij	Omogoča dodajanje in spreminjanje seznama regij	DA / NE	
Administracija mest	Omogoča dodajanje in spreminjanje seznama mest	DA / NE	
Administracija jezikov	Omogoča dodajanje in spreminjanje seznama jezikov	DA / NE	
Administracija dodatkov	Omogoča dodajanje in spreminjanje seznama dodatkov	DA / NE	
Administracija prevodov	Omogoča dodajanje in spreminjanje prevodov šifrantov	DA / NE	
Modul za registracijo	Omogoča kreiranje uporabnika na podlagi parametrov: uporabniško ime, geslo, e-pošta, varnostno vprašanje, varnostni odgovor	DA / NE	
	Pri vnosu obstoječega uporabniškega imena ali e-pošte ne dovoli kreiranj	DA / NE	
	Ne spusti registracijo če geslo ni vsaj 7 mest dolgo in vsebuje en znak, ki ni črka ali število	DA / NE	
		DA / NE	
Modul za prijavo	Omogoča prijavo uporabnika na podlagi parametrov uporabniško ime in geslo	DA / NE	
Modul za iskanje apartmajev	Ponudi na izbiro geografske regije	DA / NE	
	Ob izboru regije ponudi pripadajoča mest	DA / NE	
	Ponudi kot filter število oseb in pred akcijo iskanja preverja če je polje številka	DA / NE	
	Ponudi kot filter minimalno ceno in pred akcijo iskanja preverja če je polje številka	DA / NE	
	Ponudi kot filter maksimalno ceno in pred akcijo iskanja preverja če je polje številka	DA / NE	
Modul za prikaz seznama iskanih apartmajev	Prikaže podatke o posameznem apartmaju	DA / NE	
	Omogoča prikaz dolgega seznama na več strani	DA / NE	
Modul za prikaz posameznega apartmaja	Omogoča prikaz do 6 malih slik v enem od naslednjih formatov: jpg, png, gif.	DA / NE	
	Omogoča prikaz kontaktnih podatkov lastnika apartmaja	DA / NE	
	Omogoča prikaz tipa nastanitve	DA / NE	
	Omogoča prikaz števila ležajev in pomožnih ležajev	DA / NE	
	Omogoča prikaz seznama dodatne opreme, ki je apartma vsebuje	DA / NE	



	Omogoča prikaz cen za predsezono, sezono in posezono	DA / NE	
Modul za prikaz tri najbolj iskana mesta	Omogoča povečevanje številke popularnosti po vsakem iskanju mestu, ki je bilo izbrano	DA / NE	
	Omogoča slikovni prikaz treh mestih, ki imajo najvišje številke popularnosti	DA / NE	
Modul za administracijo apartmaja	Omogoča pošiljanje slik na strežnik v formatih jpg, gif in png poljubne velikosti. Poskrbeti mora za ustrezno zmanjšanje slike in shranjevanje slike.	DA / NE	
	Omogoča vnos kontaktnih podatkov lastnika apartmaja	DA / NE	
	Omogoča vnos tipa nastanitve	DA / NE	
	Omogoča vnos števila ležajev in pomožnih ležajev	DA / NE	
	Omogoča izbiro seznama dodatne opreme, ki je apartma vsebuje	DA / NE	
	Omogoča vnos cen za predsezono, sezono in posezono	DA / NE	

Tabela 2 - Plan prevzemnih testov

## 6 Projektno vodenje in izdelava projektnega plana

[1] Projektno vodenje je umetnosti in znanost. To je set orodij in veščin, ki nam pomagajo predvidevati in kontrolirati rezultat dela organizacije. Organizacija je vključena tudi v delo, ki ni projektno. Sam projekt ni del operacij podjetja, kot je na primer obračun plač, ampak ima za cilj dostaviti končni produkt.

### 6.1 Sistem projektnega vodenja

Dober sistem projektnega vodenja mora znati odgovoriti na naslednja vprašanja:

- Katere naloge je potrebno izvesti in v katerem vrstnem redu tako da dobimo končni produkt
- Kdaj bo katera naloga izvedena
- Kdo bo naredil nalogo
- Koliko bo naloga stala
- Kaj se bo zgodilo če nekatere naloge ne bodo izvršene v planiranem roku
- Kaj je najboljši način za prikaz projektnih detajlov osebam, ki so na kakršen koli način v projekt vključene

Dobro projektno vodenje ne garantira uspeh vsakega projekta, slabo projektno vodenje običajno pomaga v neuspehu.

#### 1. Kaj je to Projekt?

Projekt je uspešen če je končan v planiranem roku, če ne porabi več denarja kot je bilo planirano in ne nazadnje morajo biti naročniki zadovoljni z produktom, ki jim ga dostavimo.

Projekt je začasen. Trajanje projekta je lahko en teden, lahko pa tudi več let. Vsak projekt pa ima končni datum. Ni nujno da se končni datum ve na začetku projekta, ampak enkrat v toku se bo gotovo določil. Projekt za razliko od časovno neskončnih tekočih operacij podjetji, je časovno končen. Primeri za neskončne tekoče operacije so: odnosi z javnostjo, računovodstvo, zaposlovanje, itn.

Projekt se ne zgodi spontano. Potreben je plan in priprava preden se projekt zgodi.

Vsak projekt kreira unikatni produkt ali uslugo. To je rezultat projekta in razlog zakaj je bil projekt bil izveden.

#### 2. Pogled na projekt čez trikotnik čas, strošek, obseg (Slika 6-1)

Vsak projekt je časovno omejen, ima nek proračun in potrebuje neko količino dela. Torej im definiran obseg.

1. Čas

Vsaka aktivnost v projektu je časovno določena.

2. Strošek

Vsebuje stroške človeških virov, opreme in materiala uporabljenih v projektu. Strošek je tako kot čas omejen.

3. Obseg

Na obseg lahko gledamo z dva vidika. En je kot obseg produkta, drugi kot obseg projekta. Vsak uspešen projekt ustvari edinstven proizvod: konkreten predmet ali storitev. Izdelek opisuje področje uporabe, predvideno kakovost, značilnost, in funkcijo izdelka - pogosto zelo v detajle. Opis teh informacij se običajno imenuje specifikacija izdelka.



Slika 6-1 – Pogled na projekt čez trikotnik čas, strošek, obseg

## 6.2 Izdelava projektnega plana z uporabo orodja Microsoft Project

Niti najboljše orodje za upravljanje projektov na svetu ne more nikoli nadomestiti naše dobre sodbe. Vendar pravo orodje lahko pomaga in naj bi pomagalo v izvršitvi naslednjega:

- a. Spremljanju vseh informacij: o delu, času trajanja aktivnosti in količini virov, ki jih rabimo
- b. Vizualizaciji projektnega plana v znanih, dobro definiranih formatih
- c. Efektivno razporediti naloge in resurse

- d. Omogočiti izmenjavo projektnih informacij preko mreže z uporabo standardnih datotečnih formatov
- e. Komunikacijo z človeškimi viri in lastniki, medtem ko, glavna kontrola ostaja v rokah projektnega vodje

Za izdelavo projektnega plana uporabimo planiranje »od spodaj navzgor«. Seznam detajlnih aktivnosti uvozimo iz tabele »Ocena dela« in jih nato grupiramo v projektne faze.

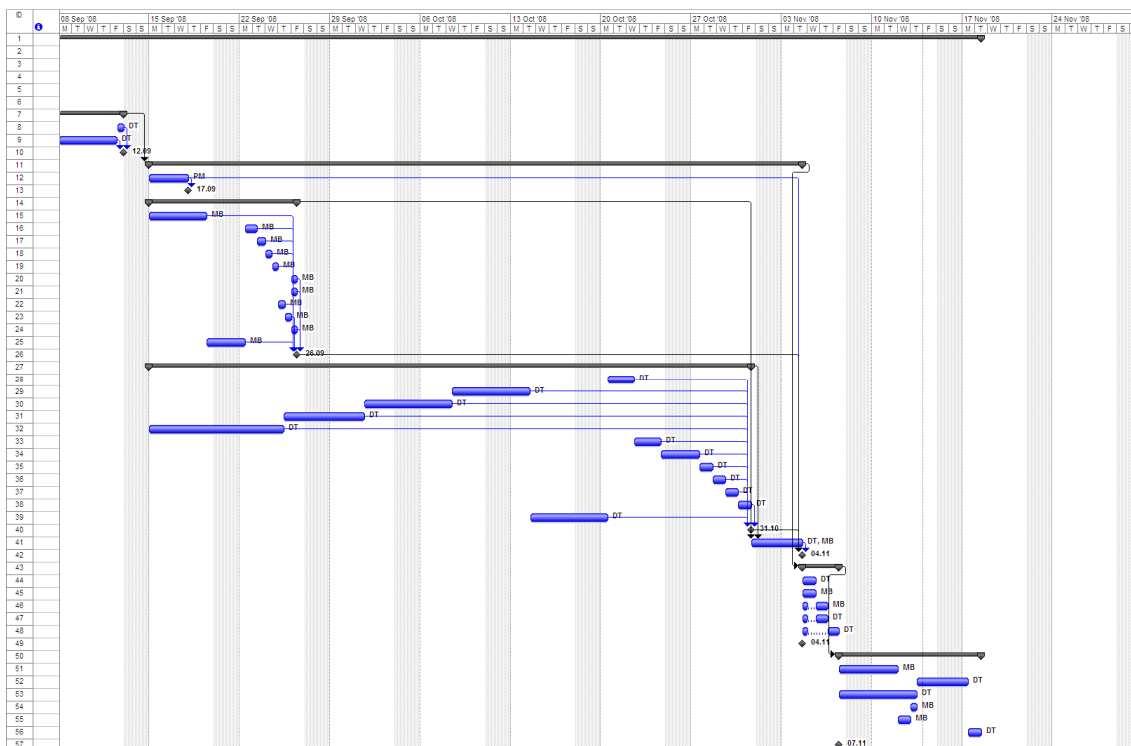
ID	Task Name	Duration	Start	Finish	Predecessors	Resource Names
1	Direktno od lastnika - Projektni plan	6.4 days?	Mon 01.09.08	Tue 18.11.08		
2						
3	<b>Analiza</b>	<b>4.5 days</b>	<b>Mon 01.09.08</b>	<b>Fri 05.09.08</b>		
4	Analiza uporabniških zahtev	3 days	Mon 01.09.08	Wed 03.09.08		DT
5	Analiza okolja	1.5 days	Thu 04.09.08	Fri 05.09.08		DT
6	Konec analize	0 days	Fri 05.09.08	Fri 05.09.08	4;5	
7	<b>Načrtovanje</b>	<b>5.5 days</b>	<b>Fri 05.09.08</b>	<b>Fri 12.09.08</b>	3	
8	Izbira arhitekture	0.5 days	Fri 12.09.08	Fri 12.09.08		DT
9	Izdelava podatkovnega modela	5 days	Fri 05.09.08	Fri 12.09.08		DT
10	Konec načrtovanja	0 days	Fri 12.09.08	Fri 12.09.08	8;9	
11	<b>Razvoj</b>	<b>36.8 days</b>	<b>Mon 15.09.08</b>	<b>Tue 04.11.08</b>	7	
12	Vzpostavitev razvojnega okolja	3 days	Mon 15.09.08	Wed 17.09.08		PM
13	Konec vzpostavitve testnega okolja	0 days	Wed 17.09.08	Wed 17.09.08	12	
14	<b>Internetne strani in oblikovanje</b>	<b>9.4 days</b>	<b>Mon 15.09.08</b>	<b>Fri 26.09.08</b>		
15	CSS oblikovanje in optimizacija slik	4.3 days	Mon 15.09.08	Fri 19.09.08		MB
16	ASP.NET master page	0.8 days	Mon 22.09.08	Tue 23.09.08		MB
17	ASP.NET stran "Iskanje namestitve"	0.8 days	Tue 23.09.08	Tue 23.09.08		MB
18	ASP.NET stran "Registracija apartmaja"	0.6 days	Tue 23.09.08	Wed 24.09.08		MB
19	ASP.NET stran "O Projektu"	0.5 days	Wed 24.09.08	Wed 24.09.08		MB
20	ASP.NET stran "Prijava"	0.3 days	Fri 26.09.08	Fri 26.09.08		MB
21	ASP.NET stran "Registracija"	0.3 days	Fri 26.09.08	Fri 26.09.08		MB
22	ASP.NET stran "Seznam apartmajev"	0.5 days	Thu 25.09.08	Thu 25.09.08		MB
23	ASP.NET stran "Pregled apartmaja"	0.5 days	Thu 25.09.08	Thu 25.09.08		MB
24	ASP.NET stran "Vnos in urejanje apartmaja"	0.4 days	Fri 26.09.08	Fri 26.09.08		MB
25	ASP.NET strani administracija šifrantov	1 day	Fri 19.09.08	Mon 22.09.08		MB
26	Konec internetne strani in oblikovanje	0 days	Fri 26.09.08	Fri 26.09.08	15;16;17;18;19;20	
27	<b>.Net</b>	<b>34.8 days</b>	<b>Mon 15.09.08</b>	<b>Fri 31.10.08</b>		
28	Modul za iskanje apartmajev	2.2 days	Mon 20.10.08	Wed 22.10.08		DT
29	Modul za prikaz seznama iskanih apartmajev	4.2 days	Wed 08.10.08	Tue 14.10.08		DT
30	Modul za prikaz posameznega apartmaja	4.5 days	Wed 01.10.08	Wed 08.10.08		DT
31	Modul za vnos in administracijo apartmajev	4.6 days	Thu 25.09.08	Wed 01.10.08		DT
32	Modul za registracijo in prijavo uporabnikov	8.2 days	Mon 15.09.08	Thu 25.09.08		DT
33	Modul za prikaz tri najbolj iskana mesta	2.1 days	Wed 22.10.08	Fri 24.10.08		DT
34	Modul za administracijo seznama jezikov	1 day	Fri 24.10.08	Mon 27.10.08		DT
35	Modul za administracijo dodatne opreme	1 day	Mon 27.10.08	Tue 28.10.08		DT
36	Modul za administracijo regij	1 day	Tue 28.10.08	Wed 29.10.08		DT
37	Modul za administracijo mest	1 day	Wed 29.10.08	Thu 30.10.08		DT
38	Module za administracijo prevodov šifrantov	1 day	Thu 30.10.08	Fri 31.10.08		DT
39	Modul za več jezikovno podporo	4 days	Tue 14.10.08	Mon 20.10.08		DT
40	Konec .Net	0 days	Fri 31.10.08	Fri 31.10.08	28;29;30;31;32;33	
41	Integracija .Net in strani	2 days	Fri 31.10.08	Tue 04.11.08	27;14	DT, MB
42	Konec razvoj	0 days	Tue 04.11.08	Tue 04.11.08	12;26;40;41	
43	<b>Sistemske testiranje</b>	<b>2.6 days?</b>	<b>Tue 04.11.08</b>	<b>Fri 07.11.08</b>	11	
44	Vzpostavitev testnega okolja	1 day?	Tue 04.11.08	Wed 05.11.08		DT
45	Priprava testnih podatkov	1 day?	Tue 04.11.08	Wed 05.11.08		MB
46	Uporabniško testiranje	1 day?	Tue 04.11.08	Thu 06.11.08		MB
47	Performance and UI Load Test	1 day?	Tue 04.11.08	Thu 06.11.08		DT
48	Bug Fixing	1 day?	Tue 04.11.08	Fri 07.11.08		DT
49	Konec sistemskega testiranja	0 days	Tue 04.11.08	Tue 04.11.08		
50	<b>Vpeljava</b>	<b>7 days</b>	<b>Fri 07.11.08</b>	<b>Tue 18.11.08</b>	43	
51	Vzpostavitev testnega okolja pri naročniku	2.5 days	Fri 07.11.08	Tue 11.11.08		MB
52	Prezemni testi	2 days	Thu 13.11.08	Mon 17.11.08		DT
53	Namestitev v produkcijsko okolje	4 days	Fri 07.11.08	Thu 13.11.08		DT
54	Izdelava namestitvene dokumentacije	0.5 days	Wed 12.11.08	Thu 13.11.08		MB
55	Uporabniški priročnik	1 day	Tue 11.11.08	Wed 12.11.08		MB
56	Predstavitve pred uporabniki	1 day	Mon 17.11.08	Tue 18.11.08		DT
57	Konec vpeljave	0 days	Fri 07.11.08	Fri 07.11.08		

Slika 6-2 - Projektne aktivnosti grupirane v projektne faze

Vsaki projektni fazi dodamo mejno aktivnost, ki označuje konec faze. Mejnimi aktivnostmi spremenimo ozadje, tako da jih lažje ločimo od ostalih aktivnosti na seznamu.

Nastavimo barvo kritičnih aktivnosti projekta kot rdečo barvo. [1] V jeziku projektnega vodenja konec projekta je določen z potjo kritičnih aktivnosti. Kritična pot je serija aktivnosti, ki bo podaljšala čas trajanja projekta če zamudimo rok izvedbe kakršne koli aktivnosti na kritični poti. Pri izvajanju projekta se je potrebno koncentrirati predvsem na izvajanje kritičnih aktivnosti.

Projektno orodje Microsoft Project nam omogoča različne grafične vpoglede na projekt. Eden izmed teh vpogledov je Ganttov diagram. Ganttov diagram je postal standardni način za vizualizacijo projektnega plana že v začetku dvajsetega stoletja. Ameriški inženir Henry Gantt je razvil diagram, ki je kazal uporabo virov v nekem obdobju. Ganttov diagram vsebuje časovno os na vrhu, ki prikazuje čas v določeni časovni enoti. Palice na grafu predstavljajo aktivnosti, tako da, začetek palice predstavlja začetek aktivnosti, konec palice konec aktivnosti. Povezave med palicami predstavljajo povezave med aktivnostmi.



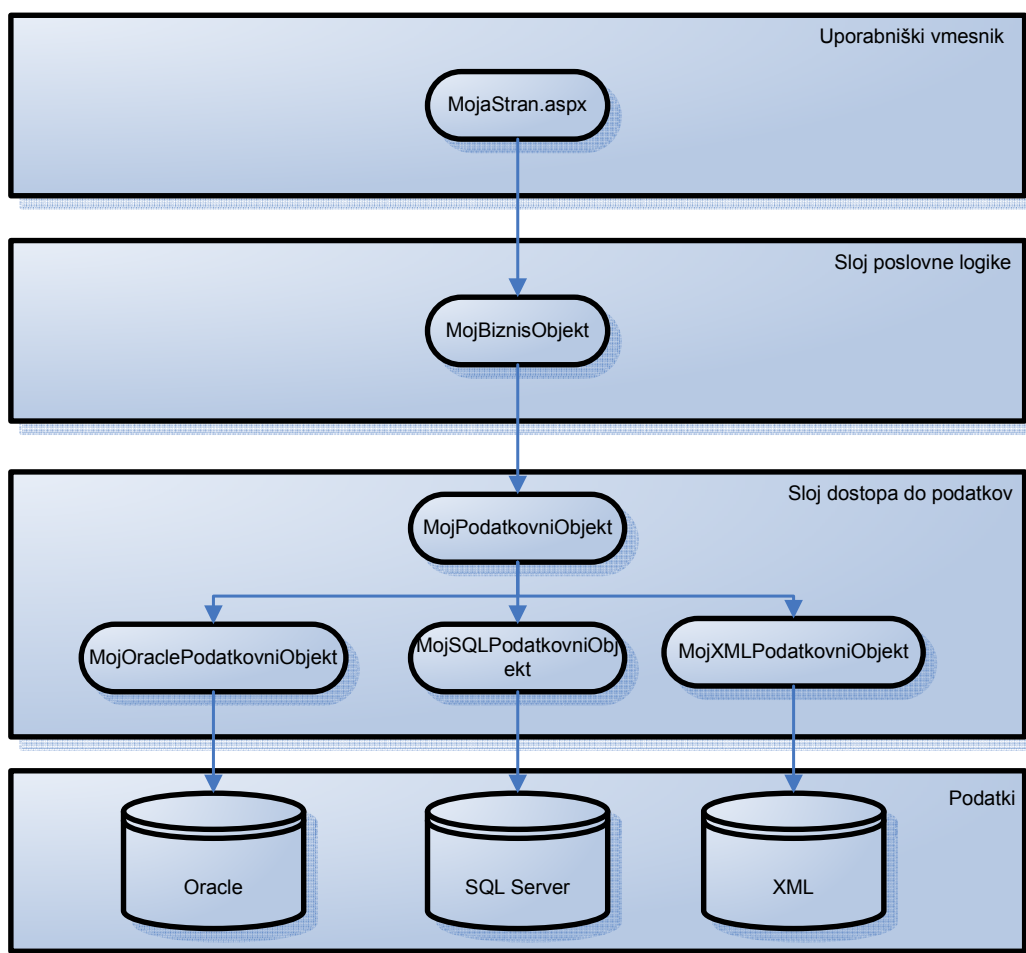
Slika 6-3 – Ganttov diagram projekta

## 7 Planiranje arhitekture aplikacije

[6] Arhitektura programske opreme določa tehnično strukturo projekta. Po dobro postavljeni arhitekturi sistema, je ostanek projekta enostaven. Slaba arhitektura naredi preostali del projekta nemogočim za izvedbo.

Naša spletna aplikacija je sestavljena iz več različnih ločenih modulov za upravljanje z dinamičnimi vsebinami. Za vse te module moramo izbrati skupni model izdelave. Odločimo se za uporabo modela večslojne arhitekture.

[2] Pri večslojni arhitekturi ločimo kodo, ki dostopa do podatkov od poslovne logike in kode uporabniškega vmesnika, tako da bo stran bolj vzdržljiva in da bo lažja implementacija dodatne zahtev v drugih fazah projekta. (Slika 7-1)



Slika 7-1 – Prikaz večslojne arhitekture

Sloj podatkov predstavlja del aplikacije, ki skrbi za shrambo podatkov. To je lahko relacijska baza, XML file, tekst file ali kakšen drug sistem primeren za podatkovno shrambo.

Sloj dostopa do podatkov predstavlja programsko kodo, ki skrbi za pridobivanje in manipuliranja podatkov, shranjenih v shrambi podatkov.

Sloj poslovne logike predstavlja programsko kodo, ki skrbi da se podatki, dobljeni od sloja dostopa do podatkov , prikažejo na bolj intuitiven način. Sloj skriva visoko raven podrobnosti, kot je podatkovna shema, in doda validacijsko logiko, ki zagotavlja, da so podatki varni in dosledni.

Sloj uporabniškega vmesnika predstavlja programsko kodo, ki definira kaj bo uporabnik videl na ekranu in na kakšen način bo prikaz formatiran.

## 8 Izdelava dizajna internetnih strani aplikacije

[4] Prvi korak pri razvoju spletne aplikacije mora biti izdelava uporabniškega dizajna. Dizajn je pomemben, ker je to tisto kaj uporabnik prvo vidi in je edini del aplikacije s katerim se sreča.

Pomembno je da je dizajn intuitiven in da sledi običajnim pravilom, na katera so uporabniki vajeni.

Pri izdelavi dizajna se trudimo narediti:

- Lepo grafični predlogo, ki zglada enako v Internet Explorer-ju in Firefox-u
- Način da z lahkoto delimo grafično predlog na vse strani, brez da bi fizično kopirali in lepili predlogo na vsako posamezno stran
- Navigacijski sistem, ki nam omogoča enostavno urejanje povezav

Za posamezno stran izdelamo grafični prikaz s orodjem za urejanje grafik, kot je Adobe-ov Photoshop (Slika 8-1).



Slika 8-1 – Izdelava grafike strani v orodju za urejanje grafike

Nato izrežemo vse slike in jih stisnemo tako da zavzamejo čim manj diskovnega prostora. S tem bomo zmanjšali internetni promet naše strani in hkrati pohitrili nalaganje strani.

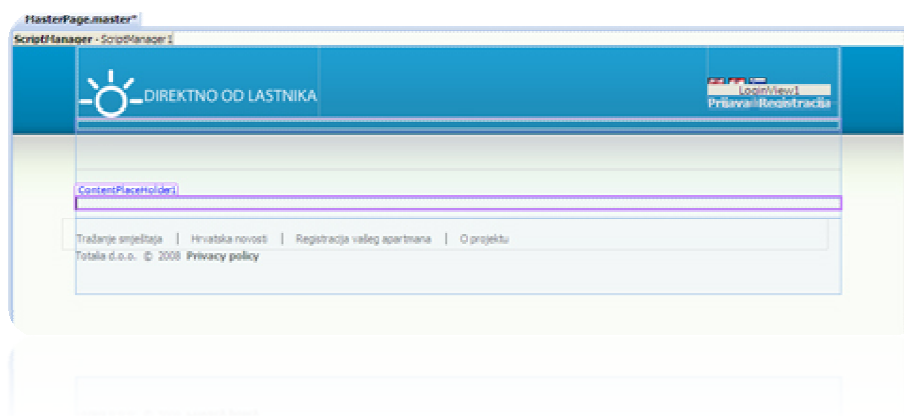
Za urejanje teksta in razporeditev elementov na straneh uporabimo tehnologijo CSS.



Tehnologija CSS nam omogoča hitro spreminjanje dizajna tudi v prihodnosti, ko bo potrebno dizajn osvežiti z novo razporeditvijo elementov ali z novimi grafičnimi elementi.

Delitev skupnega dizajna strani realiziramo z uporabo »glavne strani« (Slika 8-2). To je ASP .NET komponenta, ki omogoča definicijo skupnih področij, ki jih vsaka stran ima.

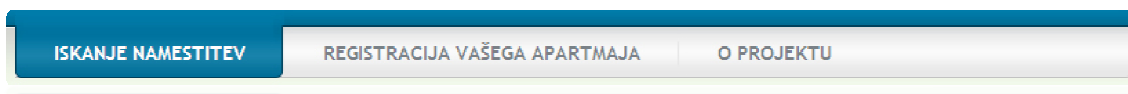
Za naše potrebe bomo izdelali dve »glavne strani«. Eno za uporabnike in eno za administratorja.



Slika 8-2 – Glavna stran aplikacije

Navigacijski sistem realiziramo s pomočjo ASP .NET »kazalo strani« komponente. »Kazalo strani« komponenta nam omogoča da v XML obliki definiramo poti do strani in naslove, ki bojo vidni na uporabniškem vmesniku.

Navigacijski sistem formatiramo na več načinov z uporabo slik in tehnologije CSS. Na ne način formatiramo prikaz v glavi strani (Slika 8-3), na drugi prikaz v nogi strani(Slika 8-4).



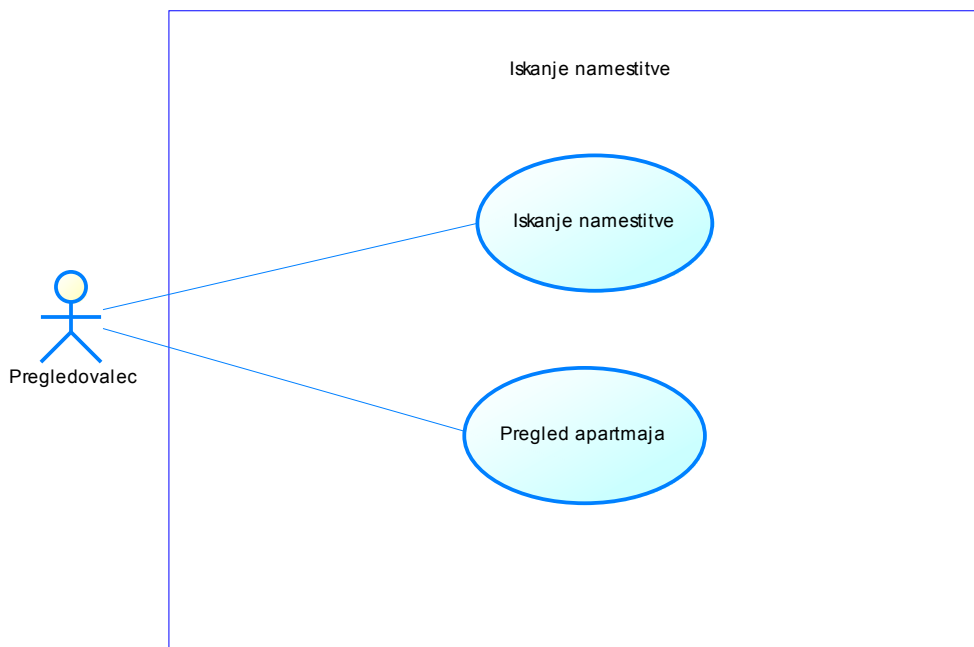
Slika 8-3 – Navigacijski sistem formatiran na način 1



Slika 8-4 - Navigacijski sistem formatiran na način 2

## 9 Modeli primerov uporabe

Pred samo izvedbo aplikacije izdelamo vse modele primerov uporabe skupaj z opisi. Primere uporabe pregledamo s naročnikom in tako odklonimo morebitne napake iz analize zahtev.



Slika 9-1 - Model primerov uporabe "Iskanje namestitve"

### 9.1 Iskanje namestitve

#### 1. Opis

Primer uporabe opisuje kako iskati namestitev.

#### 2. Tokovi dogodkov

##### 2.1. Osnovni tok

Primer uporabe se začne ko uporabnik želi iskati apartmaje.

1. Sistem zahteva vnos kriterijev za iskanje
2. Akter vpiše kriterije
3. Sistem na podlagi kriterijev preusmeri uporabnika na stran s seznamom iskalnih rezultatov

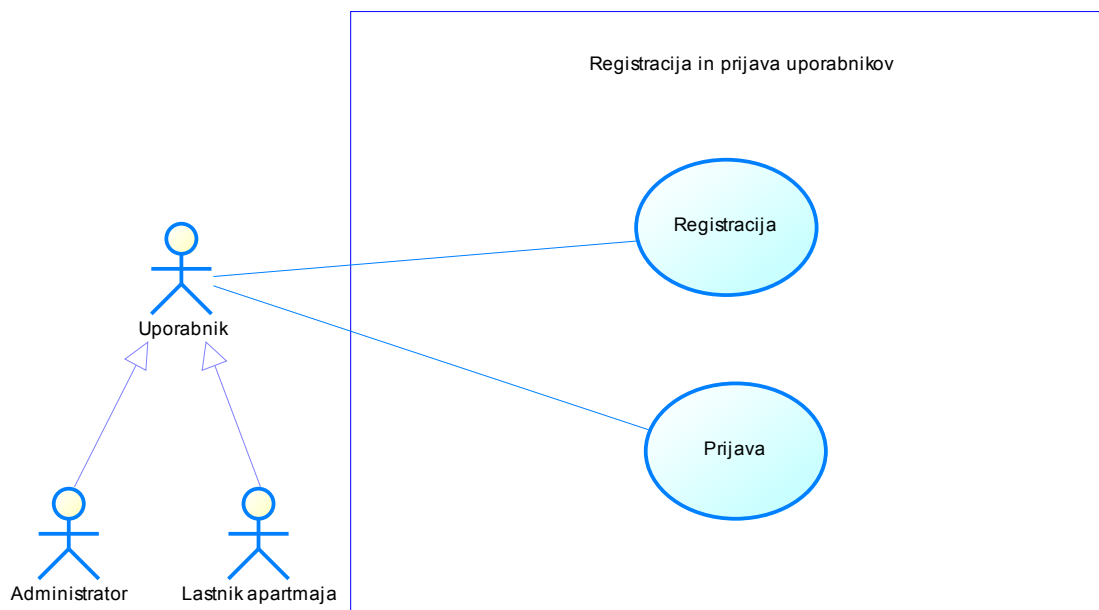
##### 2.2. Alternativni tokovi

- Napačno podani iskalni kriteriji  
Sistem opozori uporabnika če so kriteriji napačno podani

3. Zahteve  
Jih ni.
4. Predpogoji  
Jih ni.
5. Popogoji  
Jih ni.
6. Dodatne zahteve  
Jih ni.

## 9.2 Pregled apartmaja

1. Opis  
Primer uporabe opisuje kako uporabnik pregleduje posamezen apartma
2. Tokovi dogodkov
  - 2.1. Osnovni tok  
Primer uporabe se začne ko uporabnik izbere apartma za pregled iz seznama apartmajev.
    1. Sistem prikaže podatke o apartmaju
    2. Akter z premikom strani pregleda podatke
  - 2.2. Alternativni tokovi
    - Uporabnik s pošlje e-pošto lastniku apartmaja
      1. Uporabnik vnese svoje kontaktne podatke in vsebino e-pošte
      2. Sistem pošlje e-pošto na poštni naslov lastnika apartmaja
      3. Sistem obvesti uporabnika o uspešno poslani pošti
3. Zahteve  
Jih ni.
4. Predpogoji  
Seznam apartmajev vsebuje vsaj en apartma.
5. Popogoji  
Jih ni.
6. Dodatne zahteve  
Jih ni.



Slika 9-2 - Model primerov uporabe "Registracija in prijava"

### 9.3 Registracija

#### 1. Opis

Primer uporabe opisuje kako se uporabnik registrira v sistem

#### 2. Tokovi dogodkov

##### 2.1. Osnovni tok

Primer uporabe se začne ko se uporabnik želi registrirati v sistem.

1. Sistem prikaže formo za vnos podatkov o uporabniku
2. Akter vnese svoje zaželene podatke: uporabniško ime, geslo, e-pošto, varnostno vprašanje in varnostni odgovor
3. Sistem kreira novega uporabnika

##### 2.2. Alternativni tokovi

- Napačno podano ali že obstoječe uporabniško ime uporabnika
  1. Uporabnik vnese uporabniško ime, ki že obstaja ali vsebuje nedovoljene znake
  2. Sistem obvesti uporabnika o napaki

#### 3. Zahteve

Jih ni.

#### 4. Predpogoji

Uporabnik ni prijavljen v sistem.

#### 5. Popogoji

Jih ni.

## 6. Dodatne zahteve

Jih ni.

## 9.4 Prijava

### 1. Opis

Primer uporabe opisuje kako se uporabnik prijavlja v sistem

### 2. Tokovi dogodkov

#### 2.1. Osnovni tok

Primer uporabe se začne ko se uporabnik želi prijaviti v sistem

1. Sistem prikaže formo za prijavo
2. Akter se z vnosom uporabniškega ime in gesla prijavi
3. Sistem preusmeri uporabnika na stran za spremembo ali vnos podatkov o apartmaju

#### 2.2. Alternativni tokovi

- Uporabnik je administrator
  1. Akter, ki je registriran kot administrator vnese svoje uporabniške podatke
  2. Sistem preusmeri uporabnika na administratorski del aplikacije
- Napačno podani vhodni podatki
  1. Uporabnik vnese napačne podatke
  2. Sistem obvesti uporabnika o napačno vnesenih podatkih

### 3. Zahteve

Uporabnik je registriran.

### 4. Predpogoji

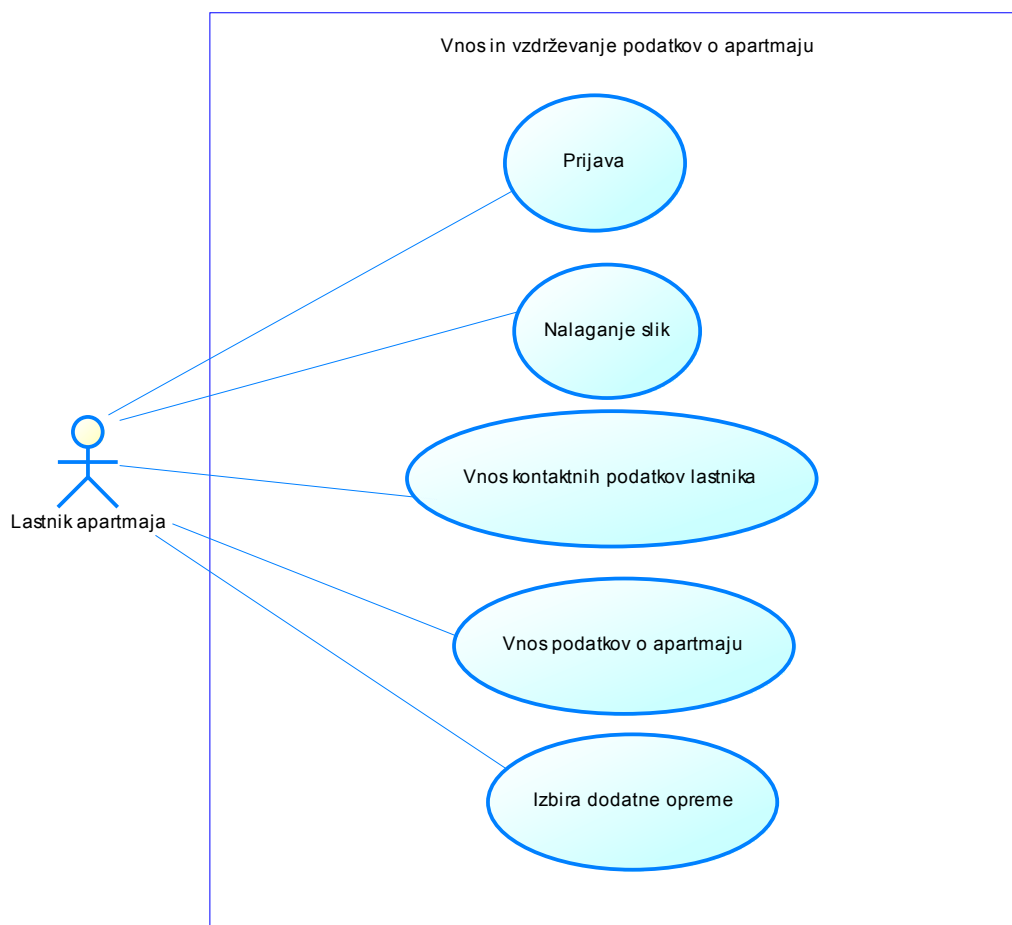
Uporabnik ni prijavljen v sistem.

### 5. Popogoji

Jih ni.

### 6. Dodatne zahteve

Jih ni.



Slika 9-3 - Model primerov uporabe "Vnos in vzdrževanje podatkov o apartmaju"

## 9.5 Nalaganje slik

### 1. Opis

Primer uporabe opisuje kako uporabnik nalaga slike

### 2. Tokovi dogodkov

#### 2.1. Osnovni tok

Primer uporabe se začne ko uporabnik klikne na gumb za nalaganje slik modula za nalaganje in pregled slik

1. Sistem prikaže vsebino datotečnega sistema uporabnika
2. Akter izbere eno sliko
3. Sistem prenese sliko z uporabnikova računalnika na strežnik

#### 2.2. Alternativni tokovi

- Uporabnik izbere datoteko, ki ni slika
  1. Sistem prikaže vsebino datotečnega sistema uporabnika

2. Akter izbere datoteko, ki ni slika
3. Sistem ne dovoli prenos datoteke, ki ni slika
3. Zahteve  
Jih ni.
4. Predpogoji  
Uporabnik je prijavljen v sistem navaden uporabnik.
5. Popogoji  
Jih ni.
6. Dodatne zahteve  
Jih ni.

## 9.6 Vnos podatkov o apartmaju

1. Opis  
Primer uporabe opisuje kako uporabnik vnaša podatke o apartmaju
2. Tokovi dogodkov
  - 2.1. Osnovni tok  
Primer uporabe se začne ko uporabnik klikne na gumb spremeni modula za vnos in spreminjanje kontaktnih podatkov apartmaja.
    1. Sistem prikaže podatke o apartmaju pripravljene za urejanje
    2. Akter vnese ali spremeni podatke in klikne na gumb shrani
    3. Sistem shrani spremenjene podatke
  - 2.2. Alternativni tokovi
    - Napačno vneseni podatki
      1. Sistem prikaže podatke o apartmaju pripravljene za urejanje
      2. Akter v polja za ceno ali število postelj vnese vrednosti, ki niso številke
      3. Sistem opozori uporabnika
3. Zahteve  
Jih ni.
4. Predpogoji  
Uporabnik je prijavljen v sistem navaden uporabnik.
5. Popogoji  
Jih ni.
6. Dodatne zahteve  
Jih ni.

## 9.7 Vnos kontaktnih podatkov lastnika

1. Opis

Primer uporabe opisuje kako uporabnik vnaša podatke o lastniku apartmaja

## 2. Tokovi dogodkov

### 2.1. Osnovni tok

Primer uporabe se začne ko uporabnik klikne na gumb spremeni modula za vnos in spreminjanje kontaktnih podatkov lastnika.

1. Sistem prikaže podatke o lastniku pripravljene za urejanje
2. Akter vnese ali spremeni podatke in klikne na gumb shrani
3. Sistem shrani spremenjene podatke

## 3. Zahteve

Jih ni.

## 4. Predpogoji

Uporabnik je prijavljen v sistem navaden uporabnik.

## 5. Popogoji

Jih ni.

## 6. Dodatne zahteve

Jih ni.

## 9.8 Izbira dodatne opreme

### 1. Opis

Primer uporabe opisuje kako uporabnik izbira dodatno opremo apartmaja

## 2. Tokovi dogodkov

### 2.1. Osnovni tok

Primer uporabe se začne ko uporabnik klikne na gumb spremeni modula za izbiro dodatne opreme apartmaja

1. Akter izbere opremo s klikom na posamezno opremo v seznamu
2. Sistem shrani izbrano opremo

## 3. Zahteve

Jih ni.

## 4. Predpogoji

Uporabnik je prijavljen v sistem navaden uporabnik.

## 5. Popogoji

Jih ni.

## 6. Dodatne zahteve

Jih ni.



## 9.9 Vzdrževanje šifrant jezikov

### 1. Opis

Primer uporabe opisuje kako uporabnik vzdržuje šifrant jezikov

### 2. Tokovi dogodkov

#### 2.1. Osnovni tok

Primer uporabe se začne ko uporabnik pride na stran za vzdrževanje šifrant jezikov

1. Sistem prikaže šifrant jezikov in formo za vnos novega jezika
2. Akter vnese podatke o novem jeziku in klikne na gumb shrani
3. Sistem shrani podatke

#### 2.2. Alternativni tokovi

- Uporabnik izbere obstoječi jezik za urejanje
  1. Akter izbere iz seznama že vnesenih jezikov en jezik
  2. Sistem omogoči spreminjanje podatkov jezika
  3. Akter klikne na gumb shrani
  4. Sistem shrani spremembe

### 3. Zahteve

Jih ni.

### 4. Predpogoji

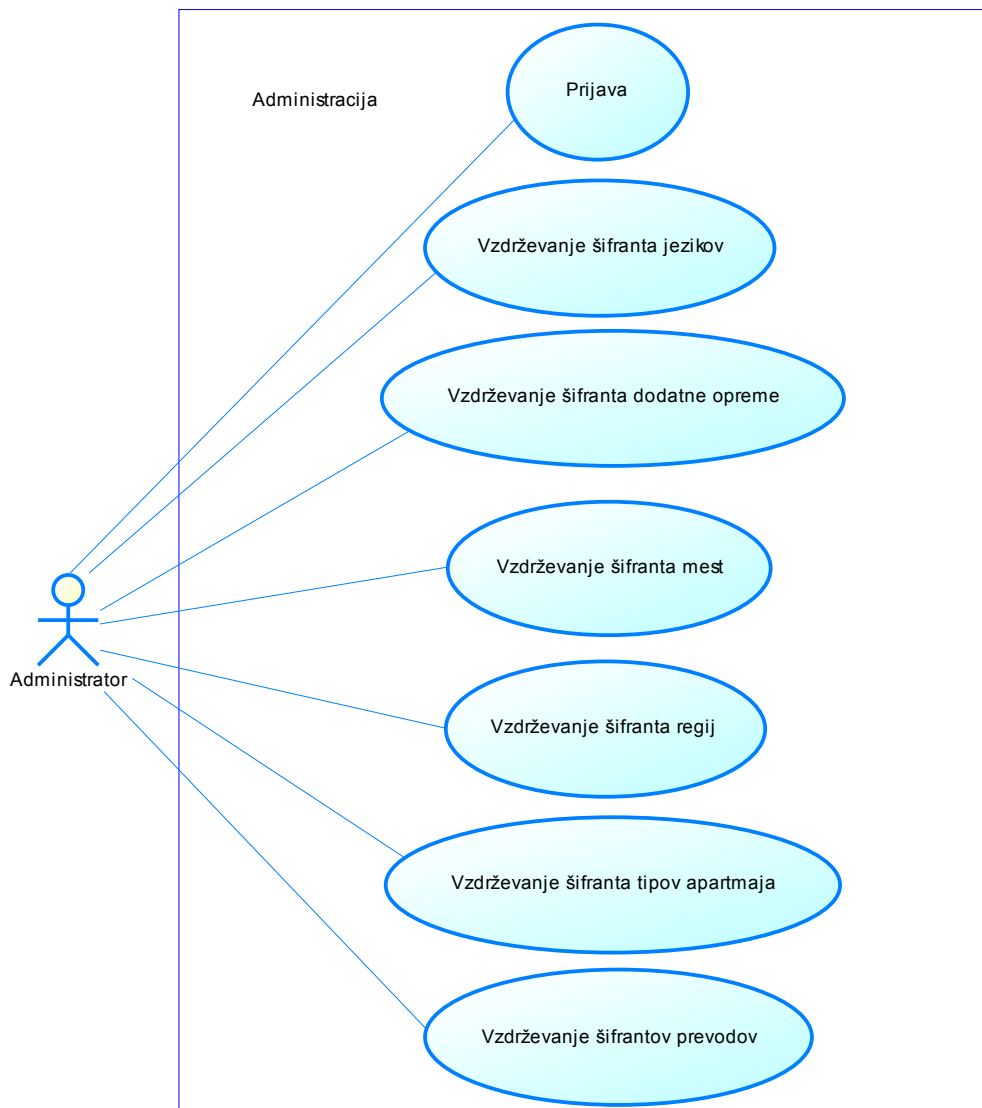
Uporabnik je prijavljen v sistem kot administrator.

### 5. Popogoji

Jih ni.

### 6. Dodatne zahteve

Jih ni.



Slika 9-4 - Model primerov uporabe "Administracija"

## 9.10 Vzdrževanje šifranta dodatne opreme

### 1. Opis

Primer uporabe opisuje kako uporabnik vzdržuje šifrant dodatne opreme

### 2. Tokovi dogodkov

#### 2.1. Osnovni tok

Primer uporabe se začne ko uporabnik pride na stran za vzdrževanje šifranta dodatne opreme

1. Sistem prikaže šifrant dodatne opreme in formo za vnos nove opreme
2. Akter vnese podatke o novi opremi in klikne na gumb shrani
3. Sistem shrani podatke

#### 2.2. Alternativni tokovi

- Uporabnik izbere obstoječi jezik za urejanje
  1. Akter izbere iz seznama že vnesenih jezikov en jezik
  2. Sistem omogoči spreminjanje podatkov jezika
  3. Akter klikne na gumb shrani
  4. Sistem shrani spremembe

### 3. Zahteve

Jih ni.

### 4. Predpogoji

Uporabnik je prijavljen v sistem kot administrator.

### 5. Popogoji

Jih ni.

### 6. Dodatne zahteve

Jih ni.

## 9.11 Vzdrževanje šifranta mest

### 1. Opis

Primer uporabe opisuje kako uporabnik vzdržuje šifrant mest

### 2. Tokovi dogodkov

#### 2.1. Osnovni tok

Primer uporabe se začne ko uporabnik pride na stran za vzdrževanje šifranta mest

1. Sistem prikaže šifranta mest in formo za vnos nove opreme
2. Akter vnese podatke o mestu in klikne na gumb shrani
3. Sistem shrani podatke

#### 2.2. Alternativni tokovi

- Uporabnik izbere obstoječe mesto za urejanje
  1. Akter izbere iz seznama že vnesenih mest eno mesto
  2. Sistem omogoči spreminjanje podatkov mesta
  3. Akter klikne na gumb shrani
  4. Sistem shrani spremembe
- 3. Zahteve  
Jih ni.
- 4. Predpogoji  
Uporabnik je prijavljen v sistem kot administrator.
- 5. Popogoji  
Jih ni.
- 6. Dodatne zahteve  
Jih ni.

## 9.12 Vzdrževanje šifrant regij

1. Opis  
Primer uporabe opisuje kako uporabnik vzdržuje šifrant regij
2. Tokovi dogodkov
  - 2.1. Osnovni tok  
Primer uporabe se začne ko uporabnik pride na stran za vzdrževanje šifrant regij
    1. Sistem prikaže šifrant regij in formo za vnos nove regije
    2. Akter vnese podatke o novi regiji in klikne na gumb shrani
    3. Sistem shrani podatke
  - 2.2. Alternativni tokovi
    - Uporabnik izbere obstoječo regijo za urejanje
      1. Akter izbere iz seznama že vnesenih regij eno regijo
      2. Sistem omogoči spreminjanje podatkov regije
      3. Akter klikne na gumb shrani
      4. Sistem shrani spremembe
3. Zahteve  
Jih ni.
4. Predpogoji  
Uporabnik je prijavljen v sistem kot administrator.
5. Popogoji  
Jih ni.
6. Dodatne zahteve  
Jih ni.

## 9.13 Vzdrževanje šifranta tipov apartmaja

### 1. Opis

Primer uporabe opisuje kako uporabnik vzdržuje šifrant tipov apartmaja

### 2. Tokovi dogodkov

#### 2.1. Osnovni tok

Primer uporabe se začne ko uporabnik pride na stran za vzdrževanje šifranta tipov apartmaja

1. Sistem prikaže šifrant tipov apartmaja in formo za vnos novega tipa apartmaja
2. Akter vnese podatke o novem tipu in klikne na gumb shrani
3. Sistem shrani podatke

#### 2.2. Alternativni tokovi

- Uporabnik izbere obstoječi tip apartmaja za urejanje
  1. Akter izbere iz seznama že vnesenih tipov apartmaja en tip
  2. Sistem omogoči spreminjanje podatkov o tipu
  3. Akter klikne na gumb shrani
  4. Sistem shrani spremembe

### 3. Zahteve

Jih ni.

### 4. Predpogoji

Uporabnik je prijavljen v sistem kot administrator.

### 5. Popogoji

Jih ni.

### 6. Dodatne zahteve

Jih ni.

## 9.14 Vzdrževanje šifranta prevodov

### 1. Opis

Primer uporabe opisuje kako uporabnik vzdržuje šifranke prevodov

### 2. Tokovi dogodkov

#### 2.1. Osnovni tok

Primer uporabe se začne ko uporabnik pride na stran za vzdrževanje šifrantov prevodov

1. Sistem prikaže šifrant prevodov regij
2. Akter vnese prevod regije in klikne na gumb shrani
3. Sistem shrani podatke

#### 2.2. Alternativni tokovi

- Uporabnik vzdržuje šifrant prevodov mest

1. Sistem prikaže seznam prevodov mest
2. Akter vnese prevod regije in klikne na gumb shrani
3. Sistem shrani podatke
- Uporabnik vzdržuje šifrant prevodov dodatne opreme
  1. Sistem prikaže seznam prevodov dodatne opreme
  2. Akter vnese prevod dodatne opreme in klikne na gumb shrani
  3. Sistem shrani podatke
3. Zahteve  
Jih ni.
4. Predpogoji  
Uporabnik je prijavljen v sistem kot administrator.
5. Popogoji  
Jih ni.
6. Dodatne zahteve  
Jih ni.

## 10 Izdelava aplikacije

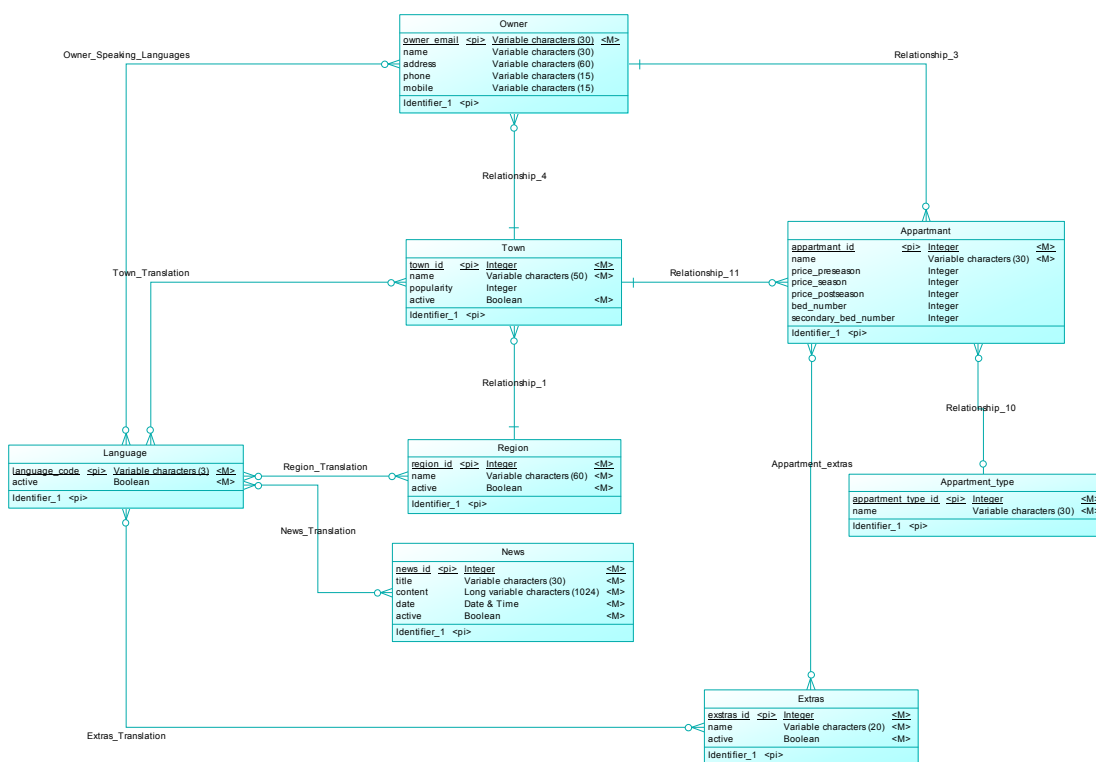
Kot smo že opredelili v zasnovi, držali se bomo večslojne arhitekture. Torej ločili bomo kodo, ki dostopa do podatkov od poslovne logike in kode uporabniškega vmesnika.

### 10.1 Sloj podatkov

Sloj podatkov predstavlja del aplikacije, ki skrbi za shrambo podatkov. V našem primeru je to relacijska podatkovna baza. V izdelavi sloja podatkov se pomagamo z »CASE<sup>5</sup>« orodjem Power Designer.

#### 10.1.1 Konceptualni podatkovni model

V konceptualnem modelu klasificiramo vse objekte v ustrezne entitete, določimo im lastnosti in jim opredelimo medsebojne relacije.



Slika 10-1 – Konceptualni podatkovni model aplikacije »Direktno od naročnikov«

<sup>5</sup> Computer-aided software engineering – programska oprema za avtomatizacijo razvoja sistemov programske opreme. Vključuje avtomatizacijo metod za oblikovanje, dokumentiranje in proizvodnjo kode v zelenem programskem jeziku.

### 10.1.1.1 Entiteta "Appartmant"

Relacija	Entiteta 2	Entiteta 1	Kardinalnost
APPARTMENT_EXTRAS	Extras	Appartmant	n,n
RELATIONSHIP_3	Appartmant	Owner	n,1
RELATIONSHIP_10	Appartmant	Appartment_type	n,1
RELATIONSHIP_11	Appartmant	Town	n,1

Tabela 3 - Seznam relacij entitete »Appartmant«

Atribut	Podatkovni tip	Obvezen
APPARTMANT_ID	Integer	X
NAME	Variable characters (30)	X
PRICE_PRESEASON	Integer	
PRICE_SEASON	Integer	
PRICE_POSTSEASON	Integer	
BED_NUMBER	Integer	
SECONDARY_BED_NUMBER	Integer	

Tabela 4 - Seznam atributov entitete »Appartmant«

### 10.1.1.2 Entiteta "Appartment\_type"

Relacija	Entiteta 2	Entiteta 1	Kardinalnost
RELATIONSHIP_10	Appartmant	Appartment_type	n,1

Tabela 5 - Seznam relacij entitete »Appartment\_type«

Atribut	Podatkovni tip	Obvezen
APPARTMENT_TYPE_ID	Integer	X
NAME	Variable characters (30)	X

Tabela 6 - Seznam atributov entitete »Appartment\_type«

### 10.1.1.3 Entiteta "Extras"

Relacija	Entiteta 2	Entiteta 1	Kardinalnost
APPARTMENT_EXTRAS	Extras	Appartmant	n,n
EXTRAS_TRANSLATION	Language	Extras	n,n

Tabela 7 - Seznam relacij entitete »Extras«

Atribut	Podatkovni tip	Obvezen
EXTRAS_ID	Integer	X
NAME	Variable characters (20)	X
ACTIVE	Boolean	X

Tabela 8 - Seznam atributov entitete »Extras«



### 10.1.1.4 Entiteta "Language"

Relacija	Entiteta 2	Entiteta 1	Kardinalnost
EXTRAS_TRANSLATION	Language	Extras	n,n
NEWS_TRANSLATION	News	Language	n,n
OWNER_SPEAKING_LANGUAGES	Language	Owner	n,n
REGION_TRANSLATION	Language	Region	n,n
TOWN_TRANSLATION	Language	Town	n,n

Tabela 9 - Seznam relacij entitete »Language«

Atribut	Podatkovni tip	Obvezen
LANGUAGE_CODE	Variable characters (3)	X
ACTIVE	Boolean	X

Tabela 10 - Seznam atributov entitete »Language«

### 10.1.1.5 Entiteta "News"

Relacija	Entiteta 2	Entiteta 1	Kardinalnost
NEWS_TRANSLATION	News	Language	0,n

Tabela 11 - Seznam relacij entitete »News«

Name	Atribut	Podatkovni tip	Obvezen
news_id	NEWS_ID	Integer	X
title	TITLE	Variable characters (30)	X
content	CONTENT	Long variable characters (1024)	X
date	DATE	Date & Time	X
active	ACTIVE	Boolean	X

Tabela 12 - Seznam atributov entitete »News«

### 10.1.1.6 Entiteta "Owner"

Relacija	Entiteta 2	Entiteta 1	Kardinalnost
OWNER_SPEAKING_LANGUAGES	Language	Owner	0,n
RELATIONSHIP_3	Appartmant	Owner	1,1
RELATIONSHIP_4	Owner	Town	1,1

Tabela 13 - Seznam relacij entitete »Owner«

Name	Atribut	Podatkovni tip	Obvezen
owner_email	OWNER_EMAIL	Variable characters (30)	X
name	NAME	Variable characters (30)	
address	ADDRESS	Variable characters (60)	
phone	PHONE	Variable characters (15)	
mobile	MOBILE	Variable characters (15)	

Tabela 14 - Seznam atributov entitete »Owner«

### 10.1.1.7 Entiteta "Region"

Relacija	Entiteta 2	Entiteta 1	Kardinalnost
REGION_TRANSLATION	Language	Region	0,n
RELATIONSHIP_1	Town	Region	1,1

Tabela 15 - Seznam relacij entitete »Region«

Atribut	Podatkovni tip	Obvezen
REGION_ID	Integer	X
NAME	Variable characters (60)	X
ACTIVE	Boolean	X

Tabela 16 - Seznam atributov entitete »Region«

### 10.1.1.8 Entiteta "Town"

Relacija	Entiteta 2	Entiteta 1	Kardinalnost
RELATIONSHIP_1	Town	Region	1,1
RELATIONSHIP_4	Owner	Town	1,1
RELATIONSHIP_11	Appartmant	Town	1,1
TOWN_TRANSLATION	Language	Town	0,n

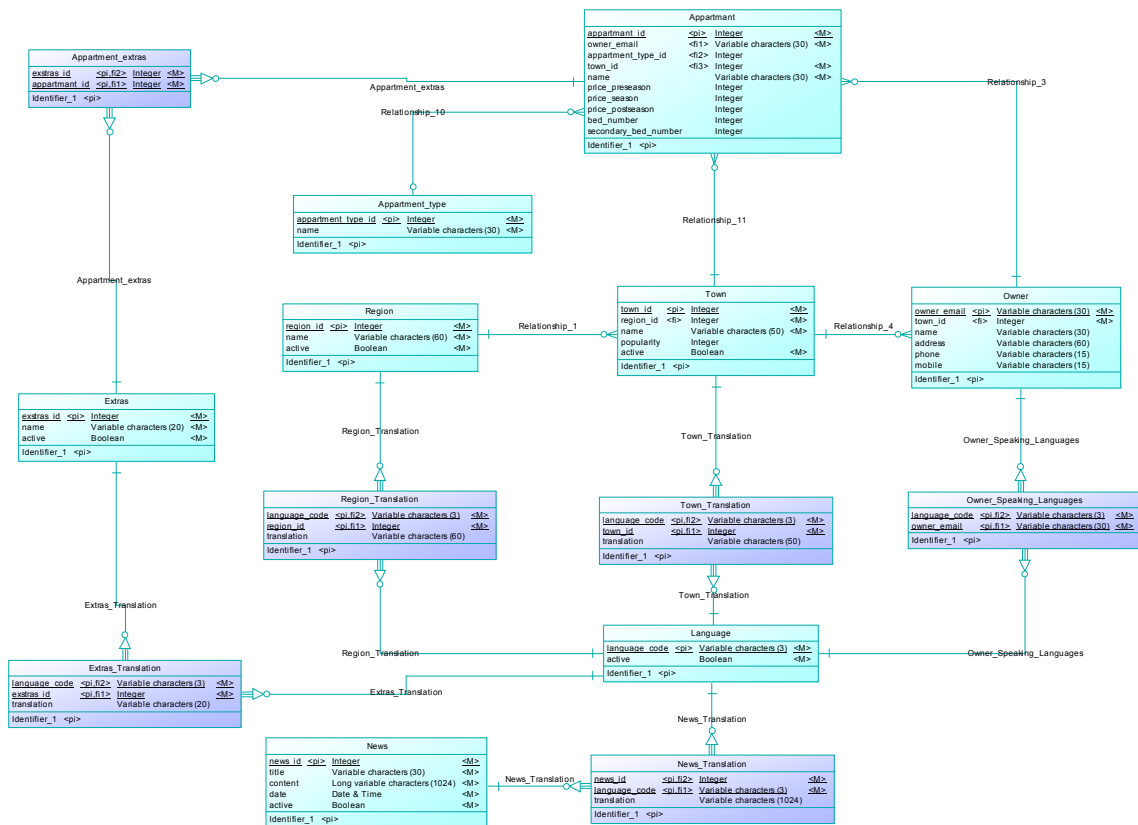
Tabela 17 - Seznam relacij entitete »Town«

Atribut	Podatkovni tip	Obvezen
TOWN_ID	Integer	X
NAME	Variable characters (50)	X
POPULARITY	Integer	
ACTIVE	Boolean	X

Tabela 18 - Seznam atributov entitete »Town«

## 10.1.2 Logični podatkovni model

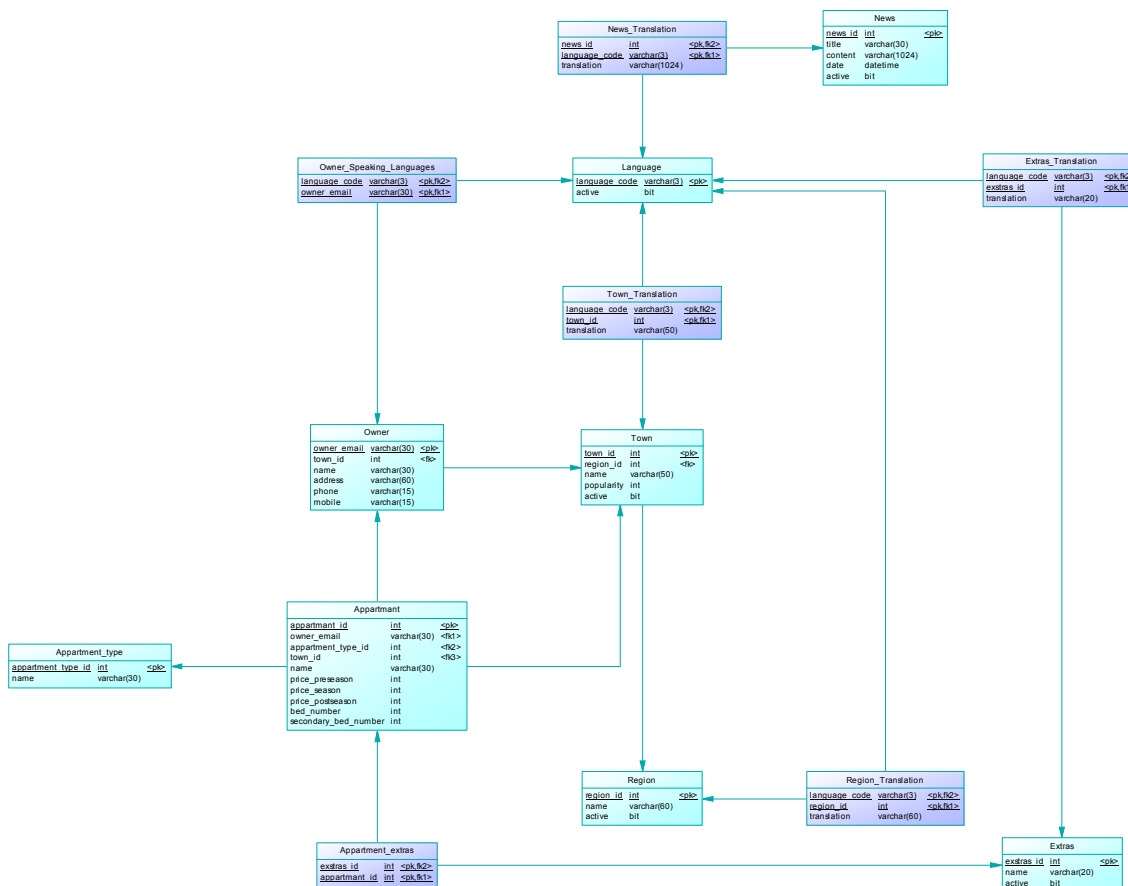
Z orodjem kreiramo iz konceptualnega modela logični model (Slika 10-2). Orodje poskrbi da se iz entitet prenesejo identifikatorji na podlagi relacij. Za vsako relacije (n,n) orodje kreira umestno entiteto (na Sliki 10-2 označene z vijolično barvo) med entitetami z to relacijo in je poveže z posamezno entiteto z relacijo (1,n). Po potrebi dodamo nove atribute novo kreiranim entitetam.



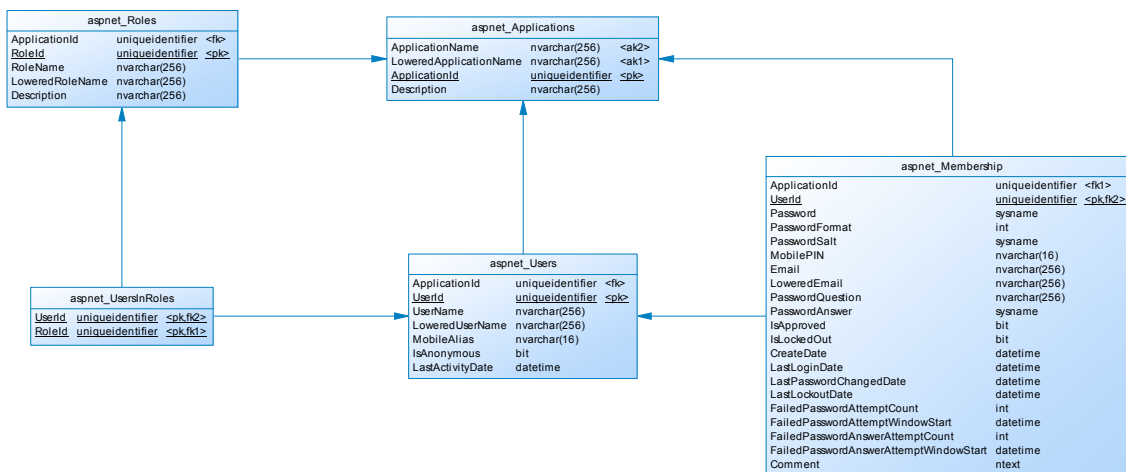
Slika 10-2 – Logični podatkovni model aplikacije »Direktno od naročnika«

### 10.1.3 Fizični podatkovni model

Prikazuje organizacijo podatkov v dejanski obliki, v kateri bodo shranjeni v bazi podatkov. Z orodjem nam je omogočen izvoz v SQL datoteko prilagojeno posamezni podatkovni bazi. SQL ukazi za kreiranje SQL Server 2005 baze so v prilogi.



Slika 10-3 – Fizični podatkovni model aplikacije »Direktno od naročnika«



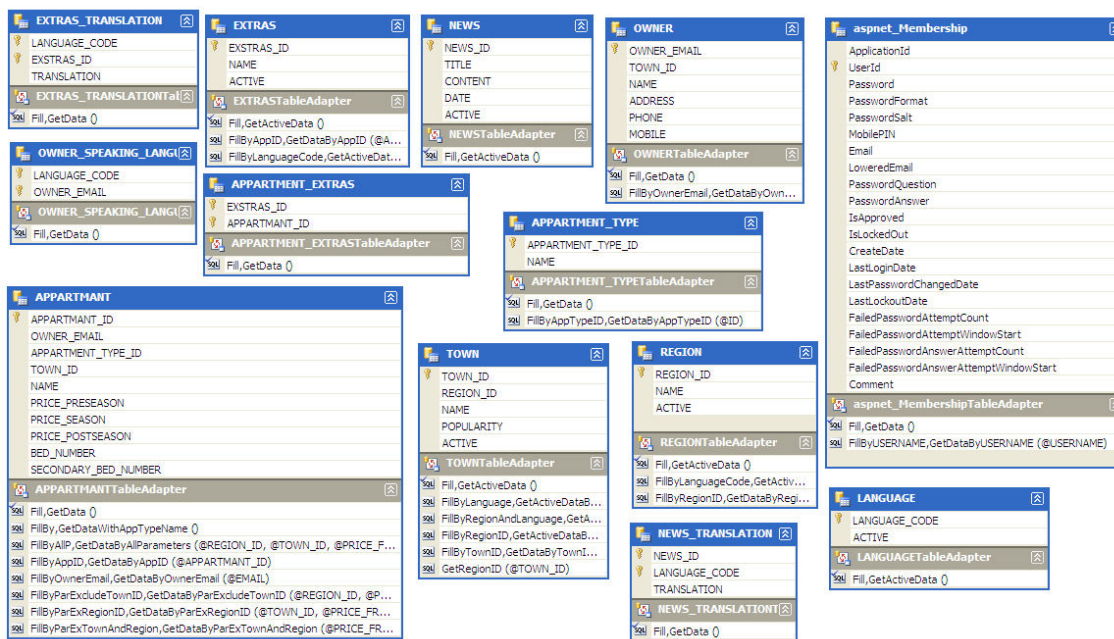
Slika 10-4 – Fizični podatkovni model modula za registracijo in prijavo uporabnikov

## 10.2 Sloj dostopa do podatkov

Sloj realiziramo z uporabo ADO.NET modela.

[7] ADO.NET je skupek knjižnic, ki so oblikovane tako, da komunicirajo z različnimi sistemi za shranjevanje podatkov. ADO.NET knjižnice omogočajo povezavo do vira podatkov, izvajanje poizvedb in obdelavo rezultatov. Zasnova ADO.NET temelji na XML in omogoča ADO.NET-u, da premosti vrzel med tradicionalnim dostopom do podatkovnih baz in XML datotekami. Kot vir podatkov razvijalci lahko uporabljajo podatkovno bazo ali XML datoteko.

Konkretna realizacija sloja je narejena preko ADO.NET objekta »DataSet«, ki ga Visual Studio zna prikazati vizualno kot diagram. Ta vizualni prikaz programske kode, ogrodje dinamično prevaja v izvorno kodo, ob dodajanju novih vizualnih elementov na diagram.



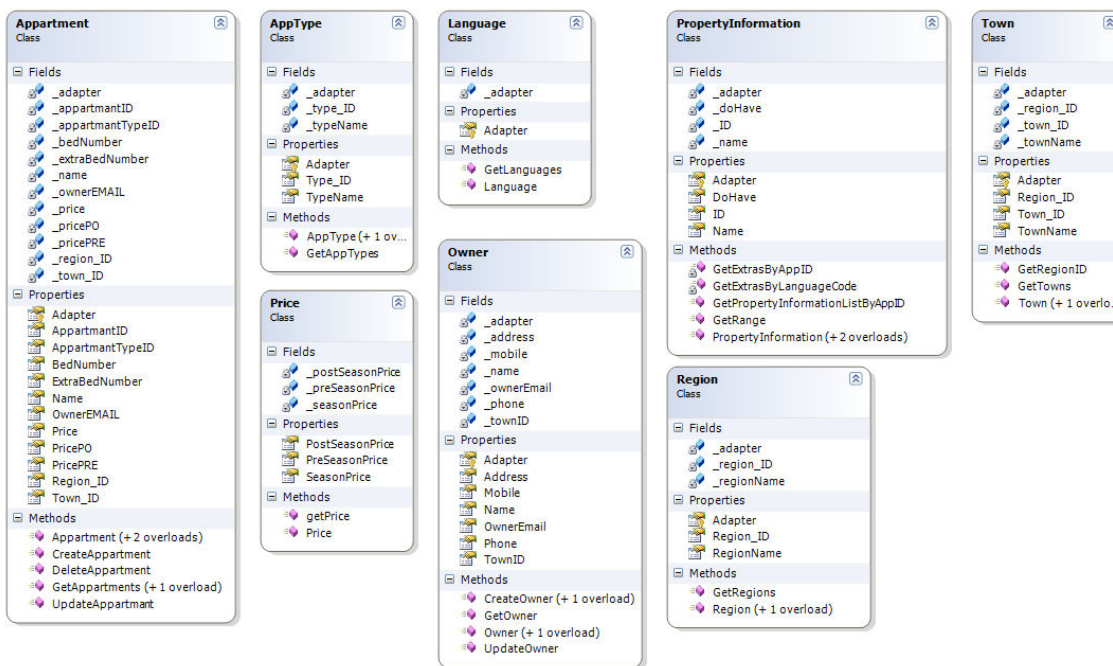
Slika 10-5 – Prikaz ADO .NET podatkovnega sloja

## 10.3 Sloj poslovne logike

Realizirati moramo entitete v objektno orientiranem načinu. Entitete so predstavljene kot razredi. Podatke dobivajo od sloja dostopa do podatkov in jih preoblikujejo na podlagi poslovnih pravil in procesov.

Za registracijo in prijavo uporabnikov bomo uporabili kontrole vgrajene v .NET ogrodje kot so: kontrola za kreacijo novega uporabnika, kontrola za prijavo registriranih uporabnikov, kontrolo za prikaz dela strani v odvisnosti od statusa prijave. Kako .NET ogrodje ponuja direktno povezavo kontrol z podatkovno bazo Microsoft SQL Server ali z Microsoft Active Directory–em ni potrebno realizirati sloja poslovne logike.

Kreiramo razredni diagram v orodju Microsoft Visual Studio in posameznim razredom določimo odgovornosti. Prikaz realizacije razreda si lahko ogledate na primeru razreda »Town« v prilogi.



Slika 10-6 – Razredni diagram sloja poslovne logike

### 10.3.1 Razred »Appartment«

Predstavlja objekt, ki nosi in skrbi za podatke o apartmajih. Definiran je na naslednji način::

Lastnost	Opis
Adapter	Objekt za dostop do sloja podatkov
AppartmentID	ID apartmaja
AppartmentTypeID	ID tipa apartmaja
BedNumber	Število postelj
ExtraBedNumber	Število dodatnih postelj
Name	Ime apartmaja
OwnerEmail	E-pošta lastnika
Price	Cena v sezoni
PricePO	Cena v posezoni
PricePRE	Cena v predsezoni
Region_ID	ID regije
Town_ID	ID mesta

Tabela 19 - Lastnosti razreda Appartment

Metoda	Opis
CreateAppartment(string)	Kreiranje apartmaja na podlagi epošte lasnika
DeleteAppartment(int)	Brisanje apartmaja na podlagi id-a apartmaja
UpdateAppartment()	Posodobitev apartmaja
GetAppartments()	Prodobitev kolekcije vseh apartmajev

Tabela 20 - Metode razreda apartment

### 10.3.2 Razred "AppType"

Predstavlja objekt, ki nosi in skrbi za podatke o tipu apartmaja. Definiran je na naslednji način:

Lastnost	Opis
Adapter	Objekt za dostop do sloja podatkov
TypeID	ID tipa
TypeName	Ime tipa

Tabela 21 - Lastnosti razreda AppType

Metoda	Opis
GetAppTypes()	Prodobitev kolekcije vseh tipov

Tabela 22 - Metode razreda AppType

### 10.3.3 Razred "Language"

Predstavlja objekt, ki nosi in skrbi za podatke o jezikih. Definiran je na naslednji način:

Lastnost	Opis
Adapter	Objekt za dostop do sloja podatkov

Tabela 23 - Lastnosti razreda Language



Metoda	Opis
GetLanguages()	Prodobitev kolekcije vseh jezikov

Tabela 24 - Metode razreda Language

### 10.3.4 Razred "Owner"

Predstavlja objekt, ki nosi in skrbi za podatke o lastniku. Definiran je na naslednji način:

Lastnost	Opis
Adapter	Objekt za dostop do sloja podatkov
Address	Adresa
Mobile	Mobilni telefon
Name	Ime
OwnerEmail	Epošta lastnika
Phone	Telefon lastnika
TownID	ID mesta lasntika

Tabela 25 - Lastnosti razreda Owner

Metoda	Opis
GetOwner	Pridobitev objekta lasntnik na podlagi epošte
CreateOwner	Kreiranje lasnitka
UpadteOwner	Posodobitev lastnika

Tabela 26 - Metode razreda Owner

### 10.3.5 Razred "PropertyInformation"

Predstavlja objekt, ki skrbi za informacije o apartmaju in je definiran na naslednji način:

Lastnost	Opis
Adapter	Objekt za dostop do sloja podatkov
DoHave	Določa če apartma vsebuje lastnost
ID	ID
Name	Ime

Tabela 27 - Lastnosti razreda PropertyInformation

Metoda	Opis
GetExtrasByAppID	Pridobitev lastnosti za posamezen apartman na podlagi ID-a
GetExtrasByLanguageCode	Pridobitev vseh informacij o apartmaju v določenem jeziku
GetPropertyInformationListByAppID	Pridobitev lastnosti za posamezen apartman na podlagi ID-a kot List<PropertyInformation>
GetRange	Pridobitev dela kolekcije

Tabela 28 - Metode razreda PropertyInformation

### 10.3.6 Razred "Region"

Predstavlja objekt, ki nosi in skrbi za podatke o regijah. Definiran je na naslednji način:

Lastnost	Opis
----------	------

Adapter	Objekt za dostop do sloja podatkov
RegionID	ID regije
Region_Name	Ime regije

Tabela 29 - Lastnosti razreda Region

Metoda	Opis
GetRegions	Pridobitev vseh objektov regij v jezui, ki je trenutno izbran

Tabela 30 - Metode razreda Region

### 10.3.7 Razred "Town"

Predstavlja objekt, ki nosi in skrbi za podatke o mestih. Definiran je na naslednji način:

Lastnost	Opis
Adapter	Objekt za dostop do sloja podatkov
RegionID	ID regije
TownID	ID mesta
TownName	Ime mesta

Tabela 31 - Lastnosti razreda Town

Metoda	Opis
GetRegionID	Pridobitev ID-a regije, ki ji mesto pripada
GetTowns	Pridobitev vseh objektov mest v jezui, ki je trenutno izbran

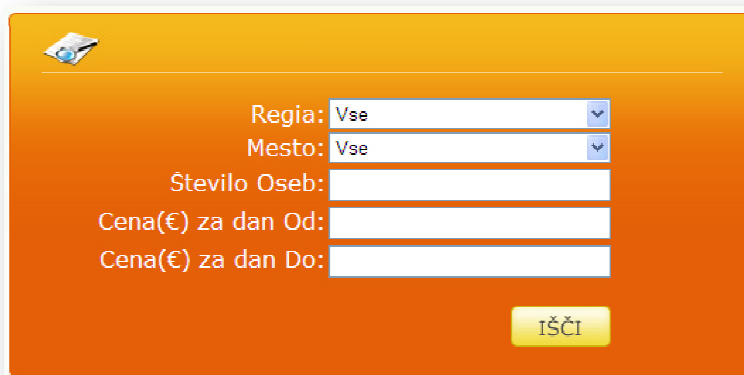
Tabela 32 - Metode razreda Town

## 10.4 Sloj uporabniškega vmesnika

Pri realizaciji uporabimo obstoječe .NET komponente za prikaz podatkov. Podatke za prikaz sloj dobi od sloja poslovne logike. Posamezne internetne strani sestavimo iz modulov, ki predstavljajo logično celino.

### 10.4.1 Modul za iskanje apartmajev

(Slika 10-7) Uporabniški vmesnik omogoča izbiro kriterijev za iskanje: regija, mesto, število oseb, minimalna cena in maksimalna cena. Na podlagi izbranih kriterijev se kliče funkcija sloja poslovne logike, ki izvrši iskanje in povrne rezultat.



Regia: Vse  
Mesto: Vse  
Število Oseb:  
Cena(€) za dan Od:  
Cena(€) za dan Do:  
IŠČI


Slika 10-7 – Uporabniški vmesnik modula za iskanje apartmajev

### 10.4.2 Modul za prikaz seznama rezultatov iskanja

(Slika 10-8) Omogoča prikaz formatiranega seznama apartmajev pridobljenega od sloja poslovne na podlagi iskalnih kriterijev. Po kliku na sliko apartmaja iz seznama, nas modul preusmeri na detajlni prikaz podatkov o apartmaju.

Apartmani Ticić				
	<b>MESTO</b> Zadar	<b>TIP</b> **	<b>ŠTEVILO POSTELJ</b> 12+2	<b>CENA</b> 5€

Apartmani Horvat				
	<b>MESTO</b> Medulin	<b>TIP</b> ****	<b>ŠTEVILO POSTELJ</b> 5+0	<b>CENA</b> 12€

Slika 10-8 – Uporabniški vmesnik modula za prikaz seznama rezultatov iskanja

### 10.4.3 Modul za registracijo uporabnika

(Slika 10-9) Omogoča vnos podatkov in obveščanje o pravilnosti podatkov pri vnosu novega uporabnika. Uporabnik lahko vnese zaželeno uporabniško ime. Če je ime že v uporabi modul izpiše obvestilo. V primeru da geslo ni dolgo 7 znakov in da ne vsebuje znaka, ki ni črka ali številka modul opozori uporabnika. Opozorilo se izpiše tudi ob nepravilnem vnosu e-pošte, ali nepravilnem vnosu ponovljenega gesla.

Uporabniško ime:

Geslo:

Ponovljeno geslo:

E-mail:

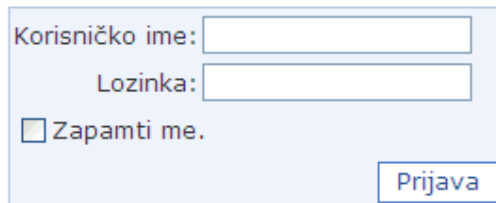
Varnostno vprašanje:

Varnostni odgovor:

Slika 10-9 - Uporabniški vmesnik modula za registracijo uporabnika

#### 10.4.4 Modul za prijavo uporabnikov v sistem

(Slika 10-10) Omogoča vnos uporabniškega imena in gesla. Skrbi da so polja za vnos izpolnjena.



Korisničko ime:

Lozinka:

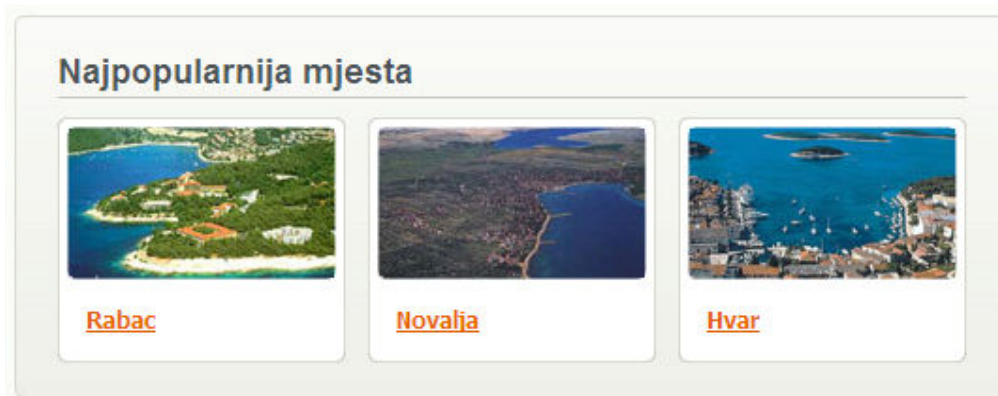
Zapamti me.

Prijava

Slika 10-10 - Uporabniški vmesnik modula za prijavo prikazan v hrvaščini

#### 10.4.5 Modul za prikaz tri najboljiskana mesta

(Slika 10-12) Omogoča prikaz slik in imen treh najboljiskanih mest.



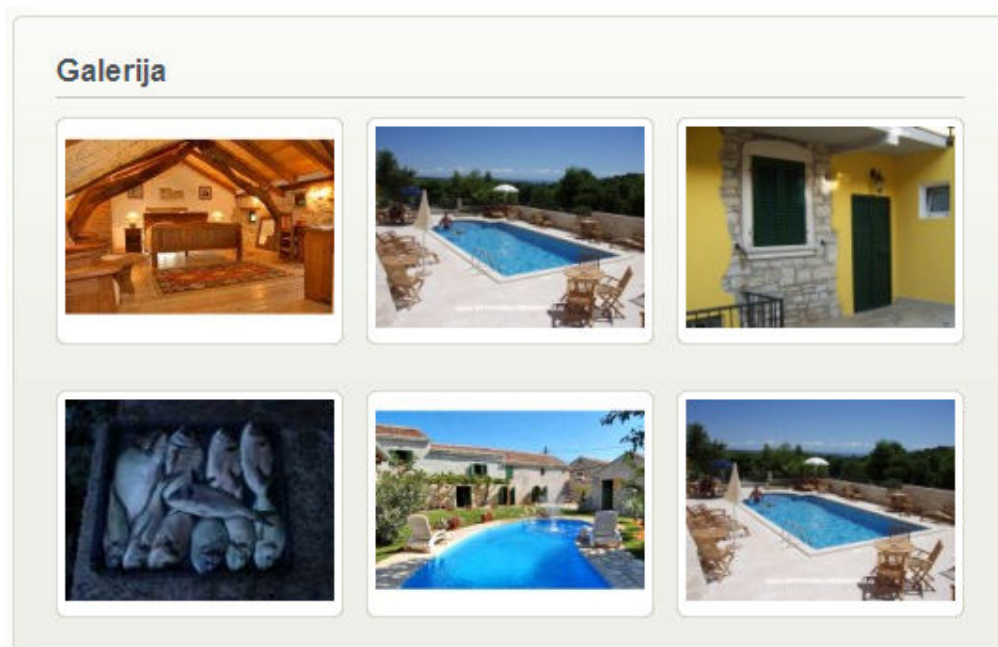
Slika 10-11 - Uporabniški vmesnik modula za prikaz tri najbolj iskanih mest

NO PICTURE

Slika 10-12 - Privzeta slika

#### 10.4.6 Modul za prikaz galerije slik

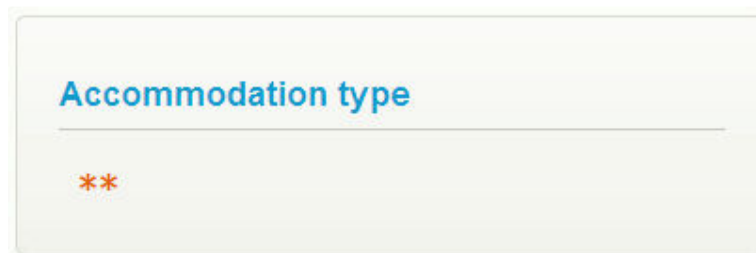
(Slika 10-13) Omogoča prikaz do 6 slik. V primeru da poti do slike ne dobi od sloja poslovne logike ali da slika ne obstaja prikaze privzeto sliko (Slika 11-10).



Slika 10-13 - Uporabniški vmesnik modula galerija

#### 10.4.7 Modul za prikaz tipa apartmaja

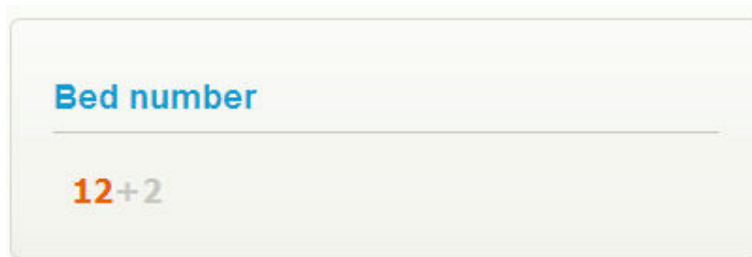
(Slika 10-14) Omogoča prikaz tipa apartmaja.



Slika 10-14 – Uporabniški vmesnik modula za prikaz tipa apartmaja

#### 10.4.8 Modul za prikaz števila postelj

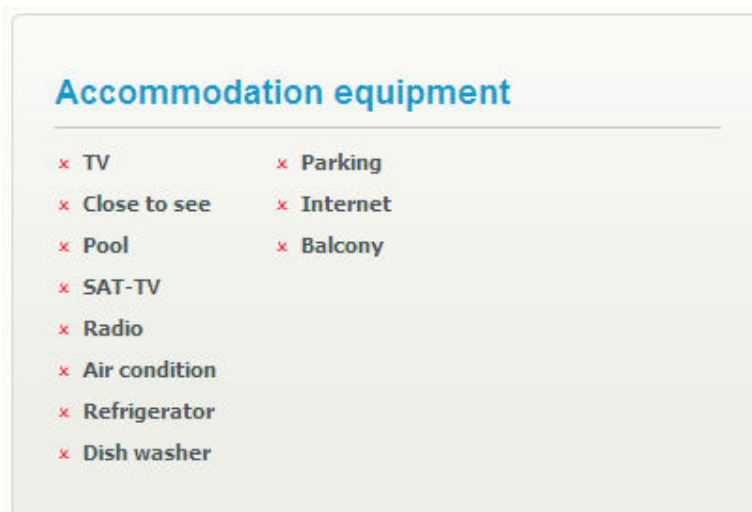
(Slika 10-15) Omogoča prikaz tipa apartmaja.



Slika 10-15 - Uporabniški vmesnik modula za prikaz števila postelj

#### 10.4.9 Modul za prikaz opreme apartmaja

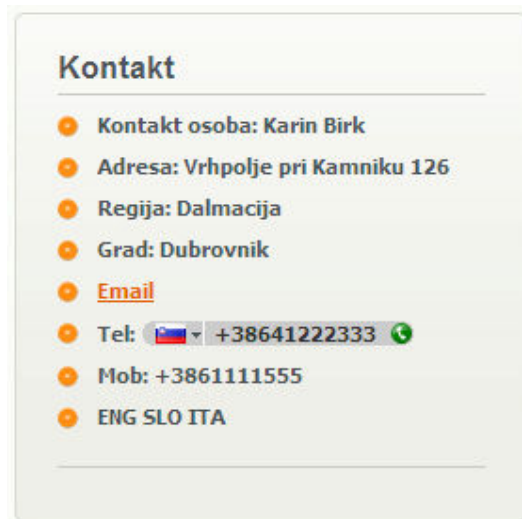
(Slika 10-16) Omogoča prikaz seznama opreme apartmaja. V primeru da apartma vsebuje opremo prikaže to opremo z zelenim križcem, če apartma ne vsebuje opreme pa z rdečim križcem.



Slika 10-16 - Uporabniški vmesnik modula za prikaz opreme apartmaja

#### 10.4.10 Modul za prikaz kontaktnih podatkov lastnika

(Slika 10-17) Omogoča prikaz kontaktnih podatkov apartmaja. Vsebuje povezavo do dela strani za pošiljanje e-pošte lastniku apartmaja. E-pošta ni eksplicitno prikazana zaradi možnih zlorab.



Slika 10-17 - Uporabniški vmesnik modula za prikaz kontaktnih podatkov lastnika

#### 10.4.11 Modul za prikaz cen po sezonah

(Slika 10-18) Omogoča prikaz cen po sezonah.

The screenshot shows a user interface titled "The price for the day". It displays prices for three seasons:

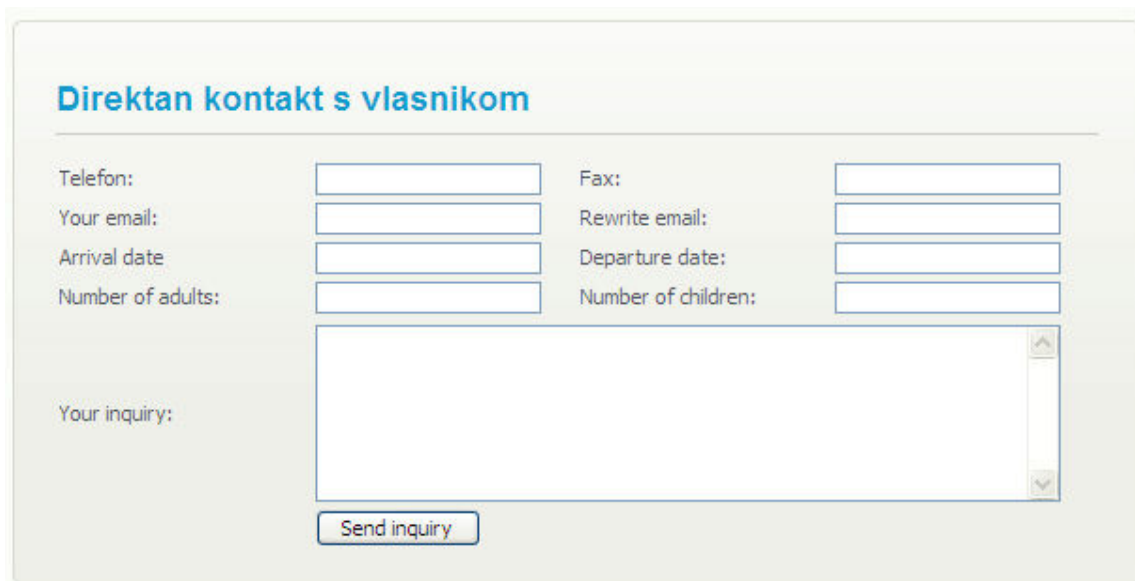
PRESESON (01.05.-01.06.)	SESON (01.06.-01.08.)	POSTSESON(01.08.-01.10.)
5	12	12

Slika 10-18 - Uporabniški vmesnik modula za prikaz cen



## 10.4.12 Modul za direkten kontakt z lastnikom

(Slika 10-19) Omogoča direkten kontakt z lastnikom. Uporabnik lahko vpiše svoje podatke: telefon, e-pošto, fax, datum prihoda, datum odhoda, število odraslih, število otrok in sporočilo.



The screenshot shows a web form titled "Direktan kontakt s vlasnikom". The form is organized into two columns of input fields. The left column contains: "Telefon:", "Your email:", "Arrival date", and "Number of adults:". The right column contains: "Fax:", "Rewrite email:", "Departure date:", and "Number of children:". Below these fields is a large text area labeled "Your inquiry:" with a vertical scrollbar on the right. At the bottom center of the form is a button labeled "Send inquiry".

Slika 10-19 - Uporabniški vmesnik modula za direkten kontakt s lastnikom

### 10.4.13 Modul za nalaganje slik v galerijo

(Slika 10-20) Omogoča nalaganje do 6 slik v galerijo slik.



The image shows a user interface for uploading pictures to a gallery. The interface is titled "Galerija" and contains six slots for pictures, arranged in two rows of three. Each slot is currently empty and contains the text "NO PICTURE". Below the text in each slot are three controls: a text input field, a "Browse..." button, and an "Upload" button. The "Browse..." button is highlighted with a red border in the top-left slot.

Slika 10-20 - Uporabniški vmesnik modula za nalaganje slik

#### 10.4.14 Modul za vpis podatkov o apartmaju

(Slika 10-21) Omogoča vpis podatkov o apartmaju: ime apartmaja, število postelj, število dodatnih postelj, cen po sezonah. Omogoča prikaz in izbiro: tipa, mesta in regije. Ob nepravilnem vnosu cene opozori uporabnika.



The image shows a web form titled "Podaci o smještaju" (Apartment Data). The form contains the following fields and controls:

- Ime apartmana:** A text input field.
- Tip apartmana:** A dropdown menu with a "\*" symbol and a downward arrow.
- Regija:** A dropdown menu with "Istra" selected and a downward arrow.
- Mesto:** A dropdown menu with "Rovinj" selected and a downward arrow.
- Broj kreveta:** A text input field with "0" entered.
- Broj dodatnih kreveta:** A text input field with "0" entered.
- Cena/Dan predsezona:** A text input field with "0" entered, followed by a "€" symbol.
- Cen/Dan sezona:** A text input field with "0" entered, followed by a "€" symbol.
- Cena/Dan posezona:** A text input field with "0" entered, followed by a "€" symbol.
- Snimi:** A button at the bottom of the form.

Slika 10-21 - Uporabniški vmesnik modula za vpis podatkov o apartmaju

### 10.4.15 Modul za vnos opreme apartmaja

(Slika 10-22) Omogoča izbiro opreme apartmaja.

Oprema Smještaja Apartman sadrži opremu(DA\NE)	
TV	<input type="checkbox"/>
SAT-TV	<input type="checkbox"/>
Blizina mora	<input type="checkbox"/>
Radio	<input type="checkbox"/>
Klima	<input type="checkbox"/>
Hladnjak	<input type="checkbox"/>
Perilica Suda	<input type="checkbox"/>
Parking	<input type="checkbox"/>
Bazen	<input type="checkbox"/>
Internet	<input type="checkbox"/>
Balkon	<input type="checkbox"/>

Snimi

Slika 10-22 - Uporabniški vmesnik modula za vnos opreme

### 10.4.16 Modul za vpis kontaktnih podatkov lastnika

(Slika 10-23) Omogoča vpis podatkov o lastniku apartmaja: ime lastnika, naslov, telefon, mobitel. Omogoča prikaz in izbiro mesta in regije, kje lastnik živi. Omogoča prikaz in dodajanje jezikov na seznam govorečih jezikov. Prikazuje e-pošto lastnika s katero se je registriral v sistem.

**Podaci o vlasniku**

Email: k5@k5.com

Ime:

Adresa:

Regija: Istra

Mjesto: Rovinj

Tel:

Mob:

Jeziki: en

Govori:

Slika 10-23 - Uporabniški vmesnik modula za vpis kontaktnih podatkov lastnika

## 11 Omejevanje dostopa do strani za registrirane uporabnike

Potrebno je zaščititi dostop do vseh strani namenjenih registriranim uporabnikom. To nam omogoča vgrajena funkcionalnost za omejevanje dostopa do map. Potrebno je v glavni konfiguracijski datoteki aplikacije, `web.config`, postaviti pravilne parametre za posamezne mape. Datoteke, ki jih želimo zaščititi pa moramo shrani znotraj teh map.

Spodnji primer prikazuje rešitev za primer naše aplikacije. Dostop do strani znotraj mape »AdminPages« imajo samo uporabniki z rolo »Administrators«, dostop do strani znotraj mape »MemberPages« pa vsi registrirani uporabniki.

Primer:

```
<configuration>
...
<location path="AdminPages">
  <system.web>
    <authorization>
      <allow roles="Administrators"/>
      <deny users="*/>
    </authorization>
  </system.web>
</location>
<location path="MemberPages">
  <system.web>
    <authorization>
      <allow users="*/>
      <deny users="?"/>
    </authorization>
  </system.web>
</location>
...
</configuration>
```

## 12 Lokalizacija strani

Zahteve nam določajo potrebo po lokalizaciji tako statičnih kot tudi dinamičnih delov naše aplikacije. Za lokalizacijo statičnih delov bomo uporabili obstoječe možnosti ASP .NET 2.0 ogrodja, za dinamične delo bomo pa razvili lastne forme za kreiranje in spreminjanje prevodov.

### 12.1 Lokalizacija statičnih elementov ASP .NET internetnih strani

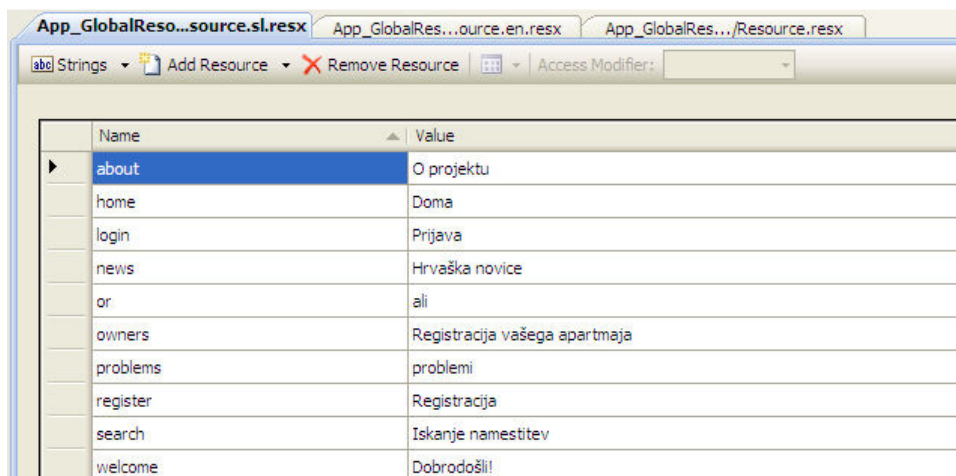
Statične elemente na internetnih straneh predstavljajo vsi teksti in vse etikete. Poskrbeti moramo da takšne elemente definiramo kot ASP .NET kontrole in jim določimo dostop do vira podatka. V spodnjem primeru smo uporabili ASP .NET kontrolo »Label« za prikaz etikete z ID-jem »Label2«. Tekst etikete pridobi iz globalnega vira s ključem »search«.

Primer:

```
<asp:Label ID="Label2" runat="server" Text = "<%$ Resources: resource,search %>">
</asp:Label>
```

[8] ASP .NET možnosti za lokalizacijo:

- a) Globalni viri (Slika 12-1) – to so file s končnico »resx« shranjeni v mapi »App\_GlobalResources«. Dinamično se prevajajo v razrede imenskega prostora »Resources«. V naši programski kodi preko teh razredov dostopamo do potrebnih podatkov. Prevode ene »resx« datoteke naredimo tako da kopiramo obstoječi in ji spremenimo ime. Ime spremenimo po pravilu »ime\_datoteke.šifra\_jezika.resx«.
- b) Lokalni viri – enako kot globalni imajo isto strukturo, končnico in način delovanja. Shranjeni so v mapi »App\_LocalResources« in dostop do posamezne »resx« datoteke ima samo stran za katero je bil vir narejen. Koristno je predvsem za vire, ki imajo isto ime (vir naslov strani) na vseh strani in jih je potrebno za vsako posamezno stran definirati na drugačen način.
- c) Avtomatično odkrivanje pravilne kulture - ogrodje .Net poskrbi za branje pravilne »resx« datotek na podlagi trenutno privzete kulture aplikacije. Ko želimo spremeniti kulturo to naredimo preko razredov »UICulture« in »CurrentUICulture«, tako da aplikaciji posodobimo kulturo in na novo naložimo stran, ki trenutno pregledujemo. Za to skrbi razred »BasePage« v prilogi.



Slika 12-1 – Globalni viri

## 12.2 Lokalizacija dinamičnih elementov

Pri lokalizaciji dinamičnih elementov mislimo na vse elemente, kateri podatki so shranjeni v bazi. To se na primer šifranti mest, regij, jezikov itn. Za lokalizacijo takšnih elementov moramo poskrbeti na vseh nivojih arhitekture. V bazi kreiramo tabele prevodov. V sloju dostopa do podatkov kreiramo metode za branje prevodov iz baze. V sloju poslovne logike poskrbimo za klicanje pravih metod sloja dostopa do podatkov na podlagi trenutno aktualne kulture aplikacije. Za posodabljanje in kreiranje prevodov kreiramo dodatne forme.

### Regions Translations

Edit Delete	LANGUAGE	CODEREGION	NAME	TRANSLATION
<a href="#">Edit</a> <a href="#">Delete</a>	en		Istra	Istria
<a href="#">Edit</a> <a href="#">Delete</a>	en		Dalmacija	Dalmatia
<a href="#">Edit</a> <a href="#">Delete</a>	en		Gorski kotar	Gorski kotar
<a href="#">Edit</a> <a href="#">Delete</a>	en		Kvarner	Kvarner
<a href="#">Edit</a> <a href="#">Delete</a>	sl		Istra	Istra
<a href="#">Edit</a> <a href="#">Delete</a>	sl		Dalmacija	Dalmacija
<a href="#">Edit</a> <a href="#">Delete</a>	sl		Kvarner	Kvarner
<a href="#">Edit</a> <a href="#">Delete</a>	sl		Gorski kotar	Gorski kotar

LANGUAGE:

REGION:

TRANSLATION:

[Insert](#)

Slika 12-2 - Forma za kreiranje in posodabljanje prevodov Regij



## 13 Postavitev aplikacije na produkcijski strežnik

Za postavitev strani na produkcijskem strežniku se moramo odločiti kako narediti naslednje:

- Izbrati vse datoteke, ki jih bomo postavili na strežnik in način kako jih bomo postavili
- Določiti način varovanja programske kode
- Postaviti okolje na produkcijske strežniku v katerem bo naša aplikacija pravilno delovala

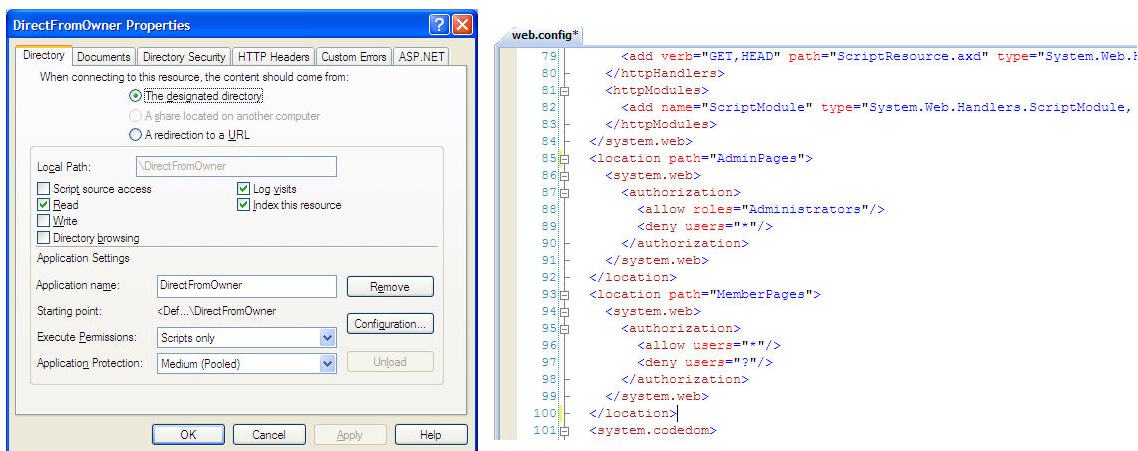
Prvi korak, ki ga moramo narediti, je preveriti verzije inštalirane programske opreme in jih po potrebi posodobiti na tiste verzije definirane v tehnični dokumentaciji.

Naslednji korak je priprava SQL skript za kreiranje podatkovne baze. Z SQL studiom izvozimo vse objekte naše podatkovne baze v eno skripto. Po želji kreiramo skripto za uvoz obstoječih podatkov v novo bazo. Zaradi uporabe vgrajenega modula za registracijo uporabnikov, uvezemo tudi to bazo. Uvoz naredimo z uporabo orodja »aspnet\_regsql.exe«.

Sledi korak prenosa vseh potrebnih datotek. Obstajata dva načina kako to lahko naredimo:

- Lahko kopiramo vse tipe datoteke takšni kot. Ta opcija je najenostavnejša vendar na ta način izvorna koda ostane v obliki navadnega teksta in je vsak z administratorskim dostopom do strežnika nemoteno bere. Za postavitve na lastne strežniku ta opcija zadošča.
- Lahko izvorno kodo prevedemo in je s tem zaščitimo. Vire kot so slike lahko prevedemo skupaj z izvorno kodo tako da jih noben ne more več naknadno spreminjati.

Končno prilagodimo postavke IIS strežnika in spremenimo našo konfiguracijsko datoteko tako da ustreza postavitvi v novem okolju (Slika 13-1).



Slika 13-1 – Konfiguracija aplikacije

## 14 Sklepne ugotovitve

Cilj diplomskega dela je bil poleg opisa vseh pomembnih korakov v izdelavi projekta ASP .NET spletne aplikacije tudi razviti konkretno aplikacijo. Del aplikacije opisan v tem delu sem uspešno razvil in potrdil s strani naročnika.

Pravilna postavitev projektih nalog in realna ocena trajanja posameznih aktivnosti mi je omogočila da projekt izpeljem brez stresa in brez podaljševanja roka zaključitve projekta. Orodje Microsoft Project nam v tem pomaga kako na začetku projekta, v vidu učinkovitega razporejanja nalog in resursov, tako tudi v toku projekt kot pomoč pri vizualizaciji projektnega plana in enostavnem spremljanju informacij o projektu.

Izbor večslojne arhitekture aplikacije se izkaže, v današnjem času ko izdelovalec programske opreme mora biti pripravljen na izdelavo dodatkov opremi in na možne spremembe po koncu projekta v vseh delih programske kode, kot edini pravilen. Njegova prednost je tudi hiter prekop aplikacije na drugo podatkovno bazo ali podatkovni vir. V mojem primeru je aplikacija napisana za bazo Microsoft SQL Server in za prekop na Oracle bi bilo potrebno spremeniti samo sloj dostopa do podatkov. V drugem primeru se lahko zgodi da uporabnik spremeni poslovna pravila, podatkovna baza pa ostane ista. Ta oblika arhitekture nam zagotavlja potrebo po spremembi, samo v sloju poslovne logike. V tretjem primeru se uporabnik lahko odloči za spremembo prikaza informacij in interakcije s aplikacijo. To rešimo s spremembo sloja uporabniškega vmesnika.

Izdelava dizajna uporabniških strani aplikacije zahteva določeno znanje s področja dizajna in izdelave grafičnih vmesnikov zato je dobro ta del prepustiti strokovnjakom za to področje. Za prikaz grafike in razporejanje grafike mi je zelo prav prišla tehnologija CSS.

Z izdelavo UML diagrama primerov uporabe in z opisi posameznih primerov smo odklonili morebitne napačne obravnave iz analize zahtev, tako da smo diagrame pregledali skupaj z naročnikom.

Za razvoj sheme podatkovne baze sem imel pomoč orodja Power Designer podjetja Sybase. Uporaba CASE orodja, ki omogoča razvoj konceptualnega modela in potem avtomatično generacijo logičnega in potem fizičnega modela, mi je omogočilo zelo hitro kreacijo fizične podatkovne baze.

Pri izdelavi aplikacije je najboljše razvojno okolje za potrebe ASP .NET razvoja definitivno Visual Studio podjetja Microsoft. Za .NET razvijalce je ono postalo standard. Z njim sem lahko urejal vse tipe datotek in diagramov, ki sem jih potreboval v razvoju, kot so: CSS datoteke, html datoteke, xsd datoteke, razredni diagram, XML diagram, itn.

Za potrebe lokalizacije uporabniških vmesnikov je tudi prav prišlo orodje Visual Studio, ki mi je omogočilo avtomatsko kreiranje datotek z lokalnimi viri in avtomatsko povezavo teh

datotek z kontrolami na formah. Za lokalizacijo šifrantov iz baze sem mogel pa sam poskrbeti z uvedbo šifrantov prevodov in razvojem metod, v vseh sloji aplikacije, ki omogočajo prikaz podatkov v jeziku, ki je trenutno izbran.

.NET vgrajene funkcionalnosti za varnost dostopa do strani so mi omogočile hitro vzpostavitev varnostne politike strani.

## Priloge

### 1. SQL ukazi za kreacijo fizične podatkovne baze

```

/*=====*/
/* DBMS name:      Microsoft SQL Server 2005      */
/* Created on:     01.10.2008 16:19:25           */
/*=====*/

/*=====*/
/* Table: APPARTMANT                               */
/*=====*/
create table APPARTMANT (
  APPARTMANT_ID      int              identity(1, 1),
  OWNER_EMAIL       varchar(30)      not null,
  APPARTMENT_TYPE_ID int              null,
  TOWN_ID           int              not null,
  NAME              varchar(30)      not null,
  PRICE_PRESEASON   int              null,
  PRICE_SEASON      int              null,
  PRICE_POSTSEASON  int              null,
  BED_NUMBER        int              null,
  SECONDARY_BED_NUMBER int          null,
  constraint PK_APPARTMANT primary key nonclustered (APPARTMANT_ID)
)
go

/*=====*/
/* Table: APPARTMENT_EXTRAS                       */
/*=====*/
create table APPARTMENT_EXTRAS (
  EXSTRAS_ID        int              not null,
  APPARTMANT_ID     int              not null,
  constraint PK_APPARTMENT_EXTRAS primary key nonclustered (EXSTRAS_ID,
APPARTMANT_ID)
)
go

/*=====*/
/* Table: APPARTMENT_TYPE                         */
/*=====*/
create table APPARTMENT_TYPE (
  APPARTMENT_TYPE_ID int              identity(1, 1),
  NAME              varchar(30)      not null,
  constraint PK_APPARTMENT_TYPE primary key nonclustered
(APPARTMENT_TYPE_ID)
)
go

/*=====*/
/* Table: EXTRAS                                  */
/*=====*/
create table EXTRAS (
  EXSTRAS_ID        int              identity(1, 1),
  NAME              varchar(20)      not null,
  ACTIVE            bit              not null,
  constraint PK_EXTRAS primary key nonclustered (EXSTRAS_ID)
)

```

```

)
go

/*=====*/
/* Table: EXTRAS_TRANSLATION */
/*=====*/
create table EXTRAS_TRANSLATION (
    LANGUAGE_CODE    varchar(3)          not null,
    EXSTRAS_ID       int                  not null,
    TRANSLATION      varchar(20)         null,
    constraint PK_EXTRAS_TRANSLATION primary key nonclustered
    (LANGUAGE_CODE, EXSTRAS_ID)
)
go

/*=====*/
/* Table: LANGUAGE */
/*=====*/
create table LANGUAGE (
    LANGUAGE_CODE    varchar(3)          not null,
    ACTIVE           bit                  not null,
    constraint PK_LANGUAGE primary key nonclustered (LANGUAGE_CODE)
)
go

/*=====*/
/* Table: NEWS */
/*=====*/
create table NEWS (
    NEWS_ID          int                  identity(1, 1),
    TITLE            varchar(30)          not null,
    CONTENT          varchar(1024)       not null,
    DATE             datetime            not null,
    ACTIVE           bit                  not null,
    constraint PK_NEWS primary key nonclustered (NEWS_ID)
)
go

/*=====*/
/* Table: NEWS_TRANSLATION */
/*=====*/
create table NEWS_TRANSLATION (
    NEWS_ID          int                  not null,
    LANGUAGE_CODE    varchar(3)          not null,
    TRANSLATION      varchar(1024)       null,
    constraint PK_NEWS_TRANSLATION primary key nonclustered (NEWS_ID,
LANGUAGE_CODE)
)
go

/*=====*/
/* Table: OWNER */
/*=====*/
create table OWNER (
    OWNER_EMAIL      varchar(30)         not null,
    TOWN_ID          int                  not null,
    NAME             varchar(30)         null,
    ADDRESS          varchar(60)         null,
    PHONE           varchar(15)         null,
    MOBILE          varchar(15)         null,
    constraint PK_OWNER primary key nonclustered (OWNER_EMAIL)
)

```

```

)
go

/*=====*/
/* Table: OWNER_SPEAKING_LANGUAGES */
/*=====*/
create table OWNER_SPEAKING_LANGUAGES (
    LANGUAGE_CODE      varchar(3)          not null,
    OWNER_EMAIL        varchar(30)         not null,
    constraint PK_OWNER_SPEAKING_LANGUAGES primary key nonclustered
(LANGUAGE_CODE, OWNER_EMAIL)
)
go

/*=====*/
/* Table: REGION */
/*=====*/
create table REGION (
    REGION_ID          int                  identity(1, 1),
    NAME               varchar(60)         not null,
    ACTIVE             bit                 not null,
    constraint PK_REGION primary key nonclustered (REGION_ID)
)
go

/*=====*/
/* Table: REGION_TRANSLATION */
/*=====*/
create table REGION_TRANSLATION (
    LANGUAGE_CODE      varchar(3)          not null,
    REGION_ID          int                 not null,
    TRANSLATION        varchar(60)         null,
    constraint PK_REGION_TRANSLATION primary key nonclustered
(LANGUAGE_CODE, REGION_ID)
)
go

/*=====*/
/* Table: TOWN */
/*=====*/
create table TOWN (
    TOWN_ID            int                  identity(1, 1),
    REGION_ID          int                 not null,
    NAME               varchar(50)         not null,
    POPULARITY         int                 null,
    ACTIVE             bit                 not null,
    constraint PK_TOWN primary key nonclustered (TOWN_ID)
)
go

/*=====*/
/* Table: TOWN_TRANSLATION */
/*=====*/
create table TOWN_TRANSLATION (
    LANGUAGE_CODE      varchar(3)          not null,
    TOWN_ID            int                 not null,
    TRANSLATION        varchar(50)         null,
    constraint PK_TOWN_TRANSLATION primary key nonclustered (LANGUAGE_CODE,
TOWN_ID)
)
go

```

## 2. XSD shema datotek za lokalizacijo

```

<xsd:schema id="root" xmlns="" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata">
  <xsd:import namespace="http://www.w3.org/XML/1998/namespace" />
  <xsd:element name="root" msdata:IsDataSet="true">
    <xsd:complexType>
      <xsd:choice maxOccurs="unbounded">
        <xsd:element name="metadata">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="value" type="xsd:string" minOccurs="0" />
            </xsd:sequence>
            <xsd:attribute name="name" use="required" type="xsd:string" />
            <xsd:attribute name="type" type="xsd:string" />
            <xsd:attribute name="mimetype" type="xsd:string" />
            <xsd:attribute ref="xml:space" />
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="assembly">
          <xsd:complexType>
            <xsd:attribute name="alias" type="xsd:string" />
            <xsd:attribute name="name" type="xsd:string" />
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="data">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="value" type="xsd:string" minOccurs="0"
msdata:Ordinal="1" />
              <xsd:element name="comment" type="xsd:string" minOccurs="0"
msdata:Ordinal="2" />
            </xsd:sequence>
            <xsd:attribute name="name" type="xsd:string" use="required"
msdata:Ordinal="1" />
            <xsd:attribute name="type" type="xsd:string" msdata:Ordinal="3"
/>
            <xsd:attribute name="mimetype" type="xsd:string"
msdata:Ordinal="4" />
            <xsd:attribute ref="xml:space" />
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="resheader">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="value" type="xsd:string" minOccurs="0"
msdata:Ordinal="1" />
            </xsd:sequence>
            <xsd:attribute name="name" type="xsd:string" use="required" />
          </xsd:complexType>
        </xsd:element>
      </xsd:choice>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>

```

### 3. Glavni razred

Vse strani izhajajo iz razreda "BasePage". Na ta način hitro integriramo ravnanje z napakami in spremembo jezika v katerem je stran prikazana.

```
public class BasePage : System.Web.UI.Page
{
    public BasePage()
    {

    }

    protected override void OnLoad(EventArgs e)
    {
        base.OnLoad(e);
    }

    protected override void InitializeCulture()
    {
        string lang = "";
        if (Session["language"] != null)
        {
            lang = (string)Session["language"];
        }

        if (lang != "")
        {
            Thread.CurrentThread.CurrentUICulture = new CultureInfo(lang);
            Thread.CurrentThread.CurrentCulture =
CultureInfo.CreateSpecificCulture(lang);
        }
        else
        {
            Thread.CurrentThread.CurrentUICulture = new CultureInfo("");
            Thread.CurrentThread.CurrentCulture =
CultureInfo.CreateSpecificCulture("");
        }
        base.InitializeCulture();
    }

    private string _lang;
    public string Lang
    {
        get { return _lang; }
    }

    protected void HandleException(Exception ex)
    {
        bool isBoolean = true;
        bool logErrors =
Boolean.TryParse(ConfigurationManager.AppSettings["LOGErrorsIntoFile"], out
isBoolean);
        if (!isBoolean)
        {
            logErrors = false;
        }
    }
}
```



```
if (logErrors)
{
    Util.LogToLogFile(ex);
}

//set exception desc to sassion parameter
Session.Add("Error", ex);
Session.Add("ErrorMessage", ex.Message);

// To let the page finish running we clear the error
Server.ClearError();

string errorPage = ConfigurationManager.AppSettings["ErrorPage"];
Response.Redirect(errorPage);
}
}
```

#### 4. Realizacija razreda "Town" sloja poslovne logike

```

using System;
using System.Collections.Generic;
using System.Web;
using DirectFromOwnerTableAdapters;

/// <summary>
/// Town
/// </summary>
public class Town
{
    private int _town_ID;
    public int Town_ID
    {
        get { return _town_ID; }
        set { _town_ID = value; }
    }
    private int _region_ID;
    public int Region_ID
    {
        get { return _region_ID; }
        set { _region_ID = value; }
    }
    private string _townName;
    public string TownName
    {
        get { return _townName; }
        set { _townName = value; }
    }

    public Town()
    {
    }

    public Town(int id)
    {
        this._town_ID = id;
        DirectFromOwner.TOWNDataTable tDT = Adapter.GetDataByTownID(id);
        if (tDT.Count > 0)
        {
            DirectFromOwner.TOWNRow tRow = tDT[0];
            this._townName = tRow.NAME;
            this._region_ID = tRow.REGION_ID;
        }
    }

    private TOWNTableAdapter _adapter = null;
    protected TOWNTableAdapter Adapter
    {
        get
        {
            if (_adapter == null)
                _adapter = new TOWNTableAdapter();

            return _adapter;
        }
    }
}

```

```

[System.ComponentModel.DataObjectMethodAttribute(System.ComponentModel.Data
ObjectMethodType.Select, true)]
public DirectFromOwner.TOWNDataTable GetTowns(int regionID)
{
    string lang = Helpers.GetCurrentLanguageCode();
    if (regionID == -1)
    {
        if (lang == "")
        {
            return Adapter.GetActiveData();
        }
        else
        {
            return Adapter.GetActiveDataByLanguage(lang);
        }
    }
    else
    {
        if (lang == "")
        {
            return Adapter.GetActiveDataByRegionID(regionID);
        }
        else
        {
            return Adapter.GetActiveDataByRegionAndLanguage(regionID,
lang);
        }
    }
}

[System.ComponentModel.DataObjectMethodAttribute(System.ComponentModel.Data
ObjectMethodType.Select, true)]
public int GetRegionID(int townID)
{
    int? i = Adapter.GetRegionID(townID);
    return (int)i;
}

```

## 5. Uporabniški vmesnik aplikacije

The screenshot shows a web browser window titled "Iskanje namestitev - Windows Internet Explorer". The address bar contains "http://localhost:DirectFromOwner/Search.aspx". The page header includes the logo "DIREKTN O OD LASTNIKA" and a "Prjava ali Registracija" link. The main navigation bar has three tabs: "ISKANJE NAMESTITEV" (selected), "REGISTRACIJA VAŠEGA APARTMAJA", and "O PROJEKTU".

The central search form is titled "Iskanje namestitev" and contains the following fields:

- Regija: Vse (dropdown menu)
- Mesto: Vse (dropdown menu)
- Število Oseb: (text input)
- Cena(€) za dan Od: (text input)
- Cena(€) za dan Do: (text input)
- IŠČI (button)

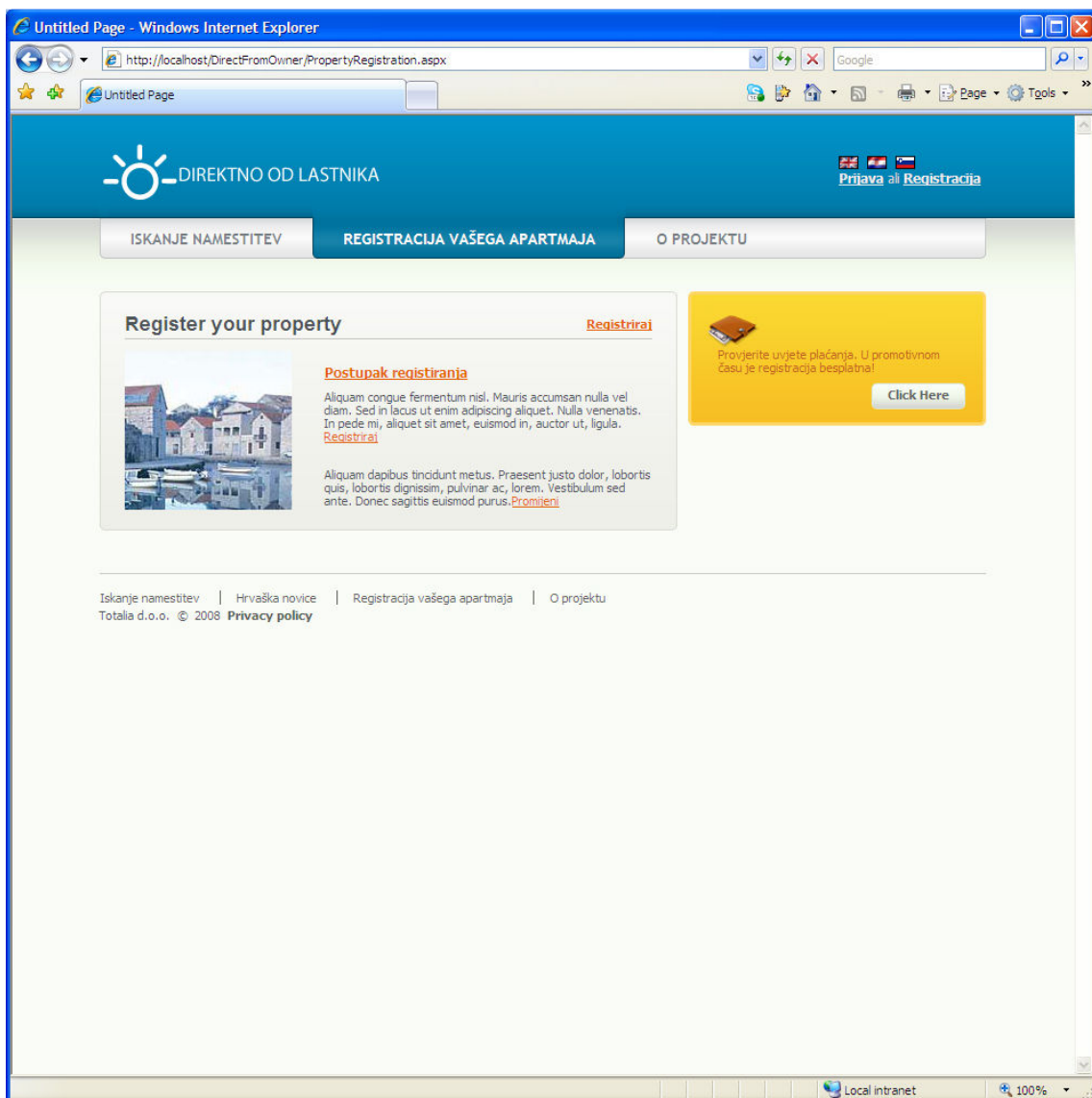
Below the search form, there is a section titled "Najbolj priljubljeni kraji" (Most popular locations) with three image thumbnails and labels: Rabac, Novalia, and Hvar.

On the right side, there is a sidebar titled "Hitra dejstva o Hrvaški" (Quick facts about Croatia) with the following numbered list:

- 1. Zemljopisni položaj:** Hrvatska se prostire od krajnjih istočnih rubov Alpa na sjeverozapadu do Panonske nizine i obala Dunava na istoku, srednji joj dio prekriva planinski masiv Dinare, a južni završava na obali Jadranskog mora.
- 2. Površina:** Kopneni dio iznosi 56.594 km<sup>2</sup>, a površina teritorijalnog mora 31.067 km<sup>2</sup>.
- 3. Naseljenost:** 4.437.460 stanovnika
- 4. Dužina morske obale:** 5.835,3 km, od toga 4.058 km obale otoka, hrđi i grebena.
- 5. Broj otoka, hrđi i grebena:** 1.244. Najveđi su otoci Krk i Cres; nastanjeni je otoka 47.
- 6. Klima:** U Hrvatskoj postoje tri klimatske zone; u unutrašnjosti prevladava umjereno topla kišna klima, na najvišim vrhuncima snježna šumska klima, a uzduž jadranske obale ugodna mediteranska klima s mnogo sunčanih dana; ljeta su suha i vruća, zime blage i vlažne. Prosječna temperatura u unutrašnjosti: siječanj od 2 do 0 °C i nešto niže na najvećim nadmorskim visinama, kolovoz oko 20 °C te oko 12 °C na najvišim vrhuncima. Prosječna temperatura u primorju: siječanj od 5 do 9 °C, kolovoz od 22 do 25 °C; temperatura mora zimi je 12 °C, a ljeti oko 25 °C.
- 7. Novac:** Službena valuta u Hrvatskoj je kuna (1 kuna = 100 lipa). Strana valuta mijenja se u bankama, njenjačnicama, poštanskim uredima i u većini turističkih agencija, hotela i kampova. Kreditne kartice (Eurocard/Mastercard, Visa, American Express i Diners) primaju se u gotovo svim hotelima, marinama, restoranima i trgovinama te na bankomatima.

The footer contains navigation links: "Iskanje namestitev", "Hrvaška novice", "Registracija vašega apartmaja", and "O projektu". It also includes the text "Totalia d.o.o. © 2008 Privacy policy" and a "Local intranet" indicator.

Slika 0-1 - Internetna stran "Iskanje namestitev"




Slika 0-2 - Internetna stran "Registracija vašega apartmaja"

Windows Internet Explorer  
http://localhost/DirectFromOwner/About.aspx

DIREKTNO OD LASTNIKA  
Prijava ili Registracija

ISKANJE NAMESTITEV | REGISTRACIJA VAŠEGA APARTMAJA | **O PROJEKTU**

### Contact Information



Totala d.o.o.  
V. i M. Lenca 67,  
Rijeka, 51000.

Telephone: +385 000 000 000  
FAX: +385 000 000 000  
E-mail: [support@totala.hr](mailto:support@totala.hr)

Proin dictum elementum velit. Fusce euismod consequat ante. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Pellentesque sed dolor. Aliquam congue fermentum nisi. Mauris accumsan nulla vel diam.

Name:

Email:

Message:

Clear Send

Iskanje namestitev | Hrvatska novice | Registracija vašega apartmaja | O projektu  
Totala d.o.o. © 2008 [Privacy policy](#)

Done Local intranet 100%

Slika 0-3 - Internetna stran "O projektu"

The screenshot shows a web browser window displaying search results for apartments. The browser's address bar shows the URL: <http://localhost/DirectFromOwner/SearchResults.aspx?rID=0&tID=0&pF=0&pT=0&nP=0>. The page header includes the logo 'DIREKTNO OD LASTNIKA' and links for 'Prijava' and 'Registracija'. Below the header, there are navigation tabs: 'ISKANJE NAMESTITEV', 'REGISTRACIJA VAŠEGA APARTMAJA', and 'O PROJEKTU'. The main content area displays three search results, each with a magnifying glass icon and a title:

- Apartmani Cervar**: Includes a photo of a building. The details are: MESTO **Lovran**, TIP **\*\*\*\***, ŠTEVILO POSTELJ **5+10**, CENA **20€**.
- Apartmani Tivic**: Includes a photo of a building entrance. The details are: MESTO **Zadar**, TIP **\*\***, ŠTEVILO POSTELJ **12+2**, CENA **5€**.
- Apartmani Horvat**: Includes a photo of a swimming pool. The details are: MESTO **Medulin**, TIP **\*\*\*\***, ŠTEVILO POSTELJ **5+0**, CENA **12€**.

The browser's status bar at the bottom shows 'Done' and 'Local intranet'.

Slika 0-4 - Internetna stran "Pregled rezultatov"

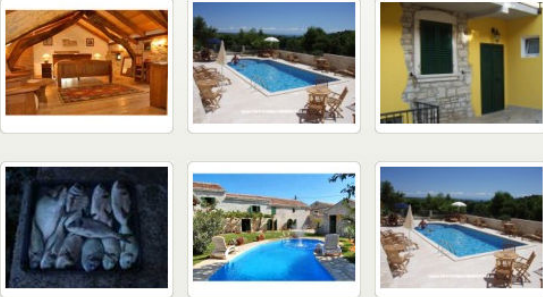
Property - Windows Internet Explorer  
 http://localhost/DirectFromOwner/Property.aspx?id=7

**DIREKTNO OD LASTNIKA** [Prijava](#) [Registracija](#)

NAZAJ NA REZULTATE ISKANJA

## Apartmani Peranic

**Galerija**



**Kontakt**

- Kontakt osoba: Luka Peranic
- Adresa:
- Regija: Dalmacija
- Grad: Zadar
- [Email](#)
- Tel:
- Mob:
- ENG SLO ITA

**Tip smještaja**

\*

**Broj ležaja**

2+0

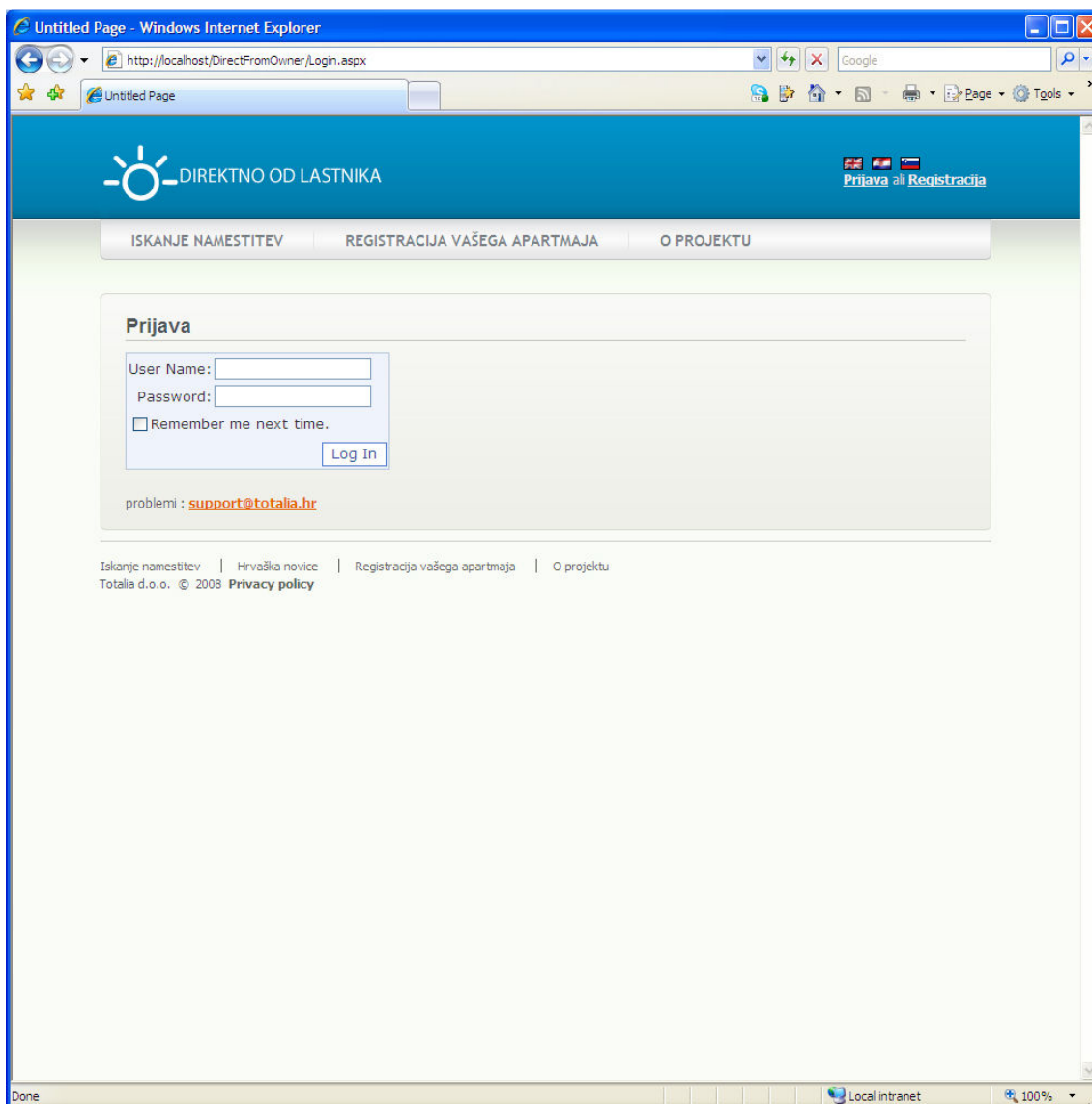
**Oprema smještaja**

- x TV
- x Bližina morja
- x Bazen
- x SAT-TV
- x Radio
- x Klima
- x Hladilnik
- x Pomivalni stroj
- x Parking
- x Balkon

Done Local intranet 100%

Slika 0-5 - Internetna stran "Pregled apartmaja"





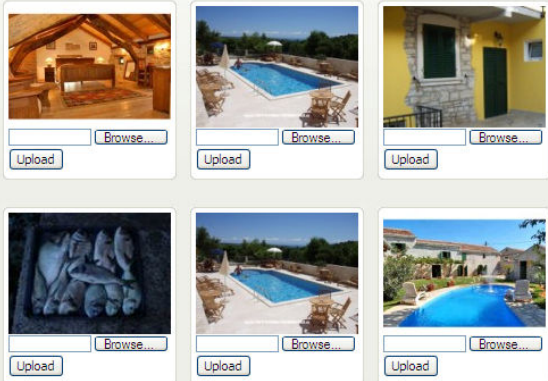
Slika 0-6 - Internetna stran "Prijava"

Untitled Page - Windows Internet Explorer  
 http://localhost/DirectFromOwner/CustomizeProperty.aspx

DIREKTNO OD VLASNIKA  
 Dobrodošli, korisnik6  
 Logout Vaš apartman

ZAVRŠI I POGLEDAJ PROMJENE

### Galerija



Šest sličica apartmana u galeriji. Svaka slika ima 'Browse...' i 'Upload' gumb.

### Podaci o vlasniku

Email: k6@k6.com

Ime: Luka Peranic

Adresa:

Regija: Istra

Mjesto: Rovinj

Tel:

Mob:

Jezik: en

Govori:

Snimi

### Podaci o smještaju

Ime apartmana: Apartmani Peranic

Tip apartmana: \*

Regija: Dalmacija

Mesto: Zadar

Broj kreveta: 2

Broj dodatnih kreveta: 0

Cena/Dan predsezona: 0 €

Cen/Dan sezona: 0 €

### Oprema smještaja

Oprema Smještaja Apartman sadrži opremu(DA/NE)

TV	<input type="checkbox"/>
SAT-TV	<input type="checkbox"/>
Blizina mora	<input type="checkbox"/>
Radio	<input type="checkbox"/>
Klima	<input type="checkbox"/>
Hladnjak	<input type="checkbox"/>
Perilica Suda	<input type="checkbox"/>
Parking	<input type="checkbox"/>
Bazen	<input type="checkbox"/>

Local intranet 100%

Slika 0-7 - Internetna stran "Spreminjanje in vnos podatkov o apartmaju"

The image shows a screenshot of a web browser window titled "Register - Windows Internet Explorer". The address bar displays "http://localhost/DirectFromOwner/Register.aspx". The browser's toolbar includes navigation buttons, a search bar with "Google", and various utility icons. The website's header is blue and features a logo with a sun icon and the text "DIREKTNO OD LASTNIKA". To the right of the logo are flags for Croatia and the text "Prijava ili Registracija". Below the header is a navigation menu with three items: "ISKANJE NAMESTITEV", "REGISTRACIJA VAŠEGA APARTMAJA", and "O PROJEKTU". The main content area is titled "Registracija" and contains a registration form with the following fields: "Uporabniško ime:", "Geslo:", "Ponovljeno geslo:", "E-mail:", "Varnostno vprašanje:", and "Varnostni odgovor:". A "Kreiraj uporabnika" button is located below the form. Below the form, there is a link for support: "Problems: [support@totalia.hr](mailto:support@totalia.hr)". At the bottom of the page, there is a footer with navigation links: "Iskanje namestitev", "Hrvaška novice", "Registracija vašega apartmaja", and "O projektu". Below these links is the text "Totalia d.o.o. © 2008 Privacy policy". The browser's status bar at the bottom shows "Done", "Local intranet", and "100%".

Slika 0-8 - Internetna stran "Registracija"

http://localhost/DirectFromOwner/AdminPages/Regions.aspx - Windows Internet Explorer

http://localhost/DirectFromOwner/AdminPages/Regions.aspx

Google

http://localhost/DirectFromOwner/AdminPages/Regio...

**DIRECT FROM OWNER** Welcome, dino [Logout](#)

**REGIONS** TOWNS LANGUAGES NEWS EXTRAS TYPES TRANSLATIONS

### Regions Administration

REGION ID	NAME	ACTIVE
<a href="#">Edit</a> <a href="#">Delete</a> 1	Istra	<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a> 2	Dalmacija	<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a> 3	Kvarner	<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a> 4	Gorski kotar	<input checked="" type="checkbox"/>

NAME:

[Insert](#)

[Search](#) [Private Accomodation](#) | [Croatia news](#) | [Register your property](#) | [About](#)  
Totalia d.o.o. © 2008 [Privacy policy](#)

Done Local intranet 100%

Slika 0-9 - Internetna stran "Administracija regij"

http://localhost/DirectFromOwner/AdminPages/Towns.aspx - Windows Internet Explorer

http://localhost/DirectFromOwner/AdminPages/Towns.aspx

http://localhost/DirectFromOwner/AdminPages/Towns...

**DIRECT FROM OWNER** Welcome, dino [Logout](#)

REGIONS **TOWNS** LANGUAGES NEWS EXTRAS TYPES TRANSLATIONS

### Towns Administration

	TOWN ID	REGION	NAME	POPULARITY	ACTIVE
<a href="#">Edit</a> <a href="#">Delete</a>	2	Istra	Rovinj		<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a>	3	Istra	Rabac		<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a>	4	Istra	Pula		<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a>	5	Istra	Medulin		<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a>	6	Istra	Fažana		<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a>	8	Dalmacija	Zadar		<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a>	9	Dalmacija	Split		<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a>	10	Dalmacija	Dubrovnik		<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a>	11	Kvarner	Rijeka		<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a>	12	Kvarner	Lovran		<input checked="" type="checkbox"/>

1 2

REGION Istra

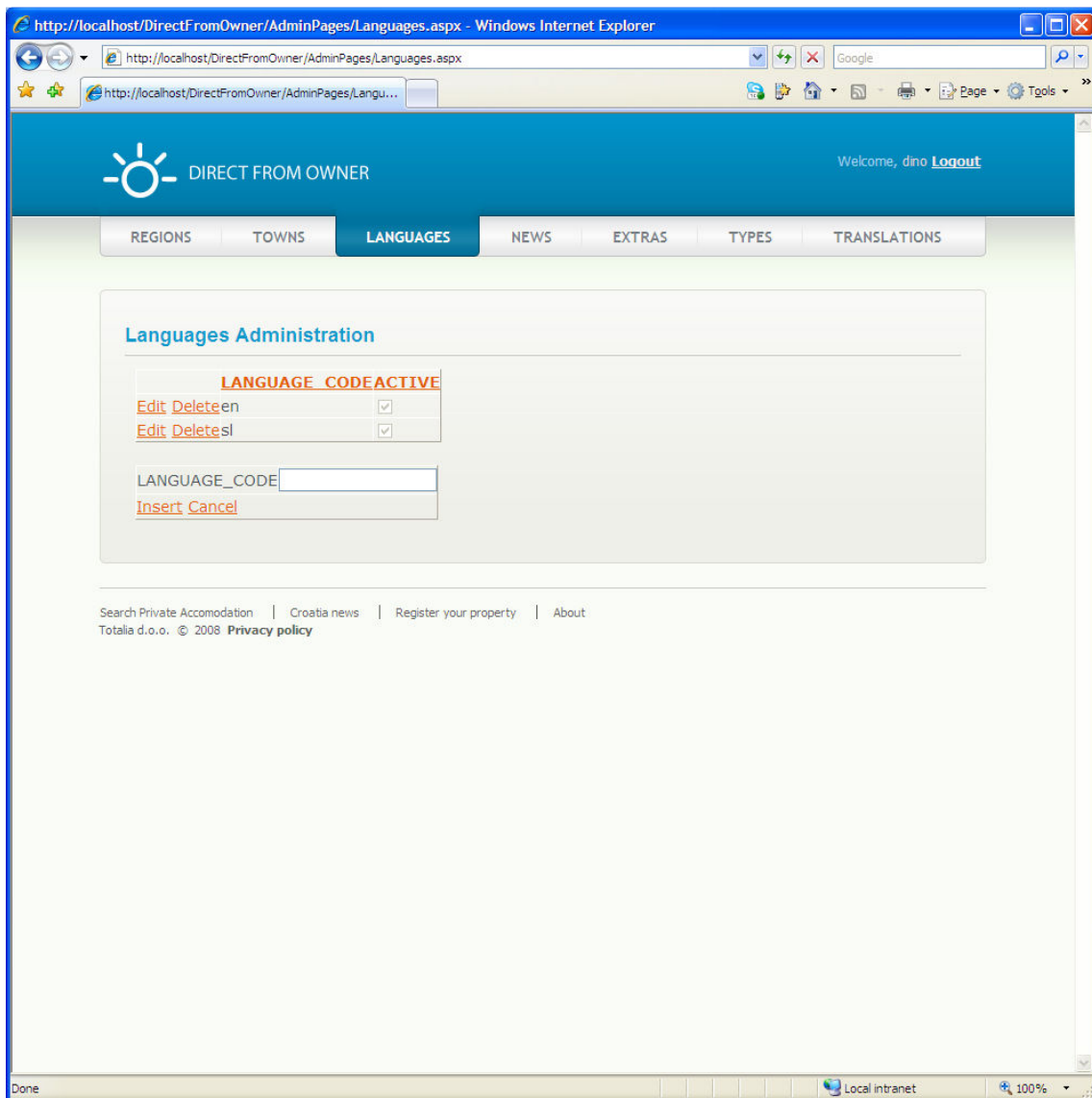
NAME

[Insert](#) [Cancel](#)

Search Private Accomodation | Croatia news | Register your property | About  
Totalia d.o.o. © 2008 [Privacy policy](#)

Done Local intranet 100%

Slika 0-10 - Internetna stran "Administracija mest"



Slika 0-11 - Internetna stran "Administracija jezikov"

The screenshot shows a web browser window displaying the 'Extras Administration' page. The page has a blue header with the logo 'DIRECT FROM OWNER' and a user greeting 'Welcome, dino Logout'. A navigation menu includes 'REGIONS', 'TOWNS', 'LANGUAGES', 'NEWS', 'EXTRAS', 'TYPES', and 'TRANSLATIONS'. The 'EXTRAS' section is active, showing a table of extras. Below the table is an 'Insert' form with a 'NAME' input field.

	EXTRAS_ID	NAME	ACTIVE
<a href="#">Edit</a> <a href="#">Delete</a>	1	TV	<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a>	2	SAT-TV	<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a>	3	Blizina mora	<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a>	11	Radio	<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a>	12	Klima	<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a>	13	Hladnjak	<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a>	14	Perilica Suda	<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a>	15	Parking	<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a>	16	Bazen	<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a>	17	Internet	<input checked="" type="checkbox"/>
<a href="#">Edit</a> <a href="#">Delete</a>	18	Balkon	<input checked="" type="checkbox"/>

NAME

[Insert](#)

Search Private Accomodation | Croatia news | Register your property | About  
Totalia d.o.o. © 2008 [Privacy policy](#)

Slika 0-12 - Internetna stran "Administracija dodatkov"

The screenshot shows a web browser window displaying the 'Types Administration' page. The browser's address bar shows the URL: `http://localhost/DirectFromOwner/AdminPages/Types.aspx`. The page has a blue header with the logo 'DIRECT FROM OWNER' and a user greeting 'Welcome, dino Logout'. Below the header is a navigation menu with tabs for 'REGIONS', 'TOWNS', 'LANGUAGES', 'NEWS', 'EXTRAS', 'TYPES' (which is active), and 'TRANSLATIONS'. The main content area is titled 'Types Administration' and contains a table with the following data:

APARTMENT TYPE IDNAME	
<a href="#">Edit</a> <a href="#">Delete</a> 5	*
<a href="#">Edit</a> <a href="#">Delete</a> 6	**
<a href="#">Edit</a> <a href="#">Delete</a> 7	***
<a href="#">Edit</a> <a href="#">Delete</a> 8	****

Below the table is a form for adding a new type, consisting of a text input field labeled 'NAME' and two buttons: 'Insert' and 'Cancel'. At the bottom of the page, there is a footer with links for 'Search Private Accomodation', 'Croatia news', 'Register your property', and 'About', along with the text 'Totalia d.o.o. © 2008 Privacy policy'. The browser's status bar at the bottom shows 'Done', 'Local intranet', and '100%' zoom.

Slika 0-13 - Internetna stran "Administracija tipov"



http://localhost/DirectFromOwner/AdminPages/Translations.aspx - Windows Internet Explorer

http://localhost/DirectFromOwner/AdminPages/Translations.aspx

http://localhost/DirectFromOwner/AdminPages/Transl...

**DIRECT FROM OWNER** Welcome, dino [Logout](#)

REGIONS TOWNS LANGUAGES NEWS EXTRAS TYPES **TRANSLATIONS**

### Regions Translations

LANGUAGE	CODE	REGION NAME	TRANSLATION
<a href="#">Edit</a> <a href="#">Delete</a>	en	Istra	Istria
<a href="#">Edit</a> <a href="#">Delete</a>	en	Dalmacija	Dalmatia
<a href="#">Edit</a> <a href="#">Delete</a>	en	Gorski kotar	Gorski kotar
<a href="#">Edit</a> <a href="#">Delete</a>	en	Kvarner	Kvarner
<a href="#">Edit</a> <a href="#">Delete</a>	sl	Istra	Istra
<a href="#">Edit</a> <a href="#">Delete</a>	sl	Dalmacija	Dalmacija
<a href="#">Edit</a> <a href="#">Delete</a>	sl	Kvarner	Kvarner
<a href="#">Edit</a> <a href="#">Delete</a>	sl	Gorski kotar	Gorski kotar

LANGUAGE: en  
 REGION: Istra  
 TRANSLATION:   
[Insert](#)

### Town Translations

LANGUAGE	CODE	TOWN NAME	TRANSLATION
<a href="#">Edit</a> <a href="#">Delete</a>	sl	Pula	Pulj
<a href="#">Edit</a> <a href="#">Delete</a>	en	Rovinj	Rovinj
<a href="#">Edit</a> <a href="#">Delete</a>	en	Rabac	Rabac
<a href="#">Edit</a> <a href="#">Delete</a>	sl	Rabac	Rabac
<a href="#">Edit</a> <a href="#">Delete</a>	en	Split	Split
<a href="#">Edit</a> <a href="#">Delete</a>	en	Pula	Pula
<a href="#">Edit</a> <a href="#">Delete</a>	sl	Rovinj	Rovinj
<a href="#">Edit</a> <a href="#">Delete</a>	sl	Split	Split

Done Local intranet 100%

Slika 0-14 - Internetna stran "Administracija prevodov"

## Viri (Literatura)

- [1] Carl Chatfield, Timothy Johnson, "Microsoft Office Project 2007 Step by Step", Redmond: Microsoft Press, 2007.
- [2] Christian Gross, "Foundations of Object-Oriented Programming Using .NET 2.0 Patterns", Berkeley: Apress, 2006.
- [3] Ken Schwaber, "Agile Project Management With Scrum", Redmond: Microsoft Press, 2004.
- [4] Marco Bellinaso, "ASP.NET 2.0 Website Programming Problem-Design-Solution", Indianapolis: Willey Publishing, Inc., 2006.
- [5] Sam Guckenheimer, Juan J. Perez, "Software Engineering with Microsoft Visual Studio Team System", Boston: Addison-wesley, 2006.
- [6] Steve McConnell, »Software Project Survival Guide«, Redmond: Microsoft Press, 1998.
- [7] (2008) ADO .NET. Dostopno na: <http://en.wikipedia.org/wiki/ADO.NET>
- [8] (2008) AJAX. Dostopno na <http://en.wikipedia.org/wiki/AJAX>
- [9] (2008) ASP .NET. Dostopno na <http://en.wikipedia.org/wiki/ASP.NET>
- [10] (2008) ASP .NET Globalization and Localization. Dostopno na: <http://msdn.microsoft.com/en-us/library/c6zyy3s9.aspx>
- [11] (2008) ASP .NET Web applications. Dostopno na: <http://msdn.microsoft.com/en-us/library/ms644563.aspx>
- [12] (2008) Cascading Style Sheets. Dostopno na: <http://en.wikipedia.org/wiki/Css>
- [13] (2008) Internet Information Services. Dostopno na: [http://en.wikipedia.org/wiki/Internet\\_Information\\_Services](http://en.wikipedia.org/wiki/Internet_Information_Services)
- [14] (2008) Microsoft Project. Dostopno na: [http://en.wikipedia.org/wiki/Microsoft\\_Project](http://en.wikipedia.org/wiki/Microsoft_Project)
- [16] (2008) Microsoft SQL Server. Dostopno na: [http://en.wikipedia.org/wiki/Microsoft\\_SQL\\_Server](http://en.wikipedia.org/wiki/Microsoft_SQL_Server)
- [17] (2008) Microsoft Visual Studio. Dostopno na: [http://en.wikipedia.org/wiki/Visual\\_studio](http://en.wikipedia.org/wiki/Visual_studio)
- [18] (2008) Microsoft .NET Framework 3.5. Dostopno na: [http://en.wikipedia.org/wiki/.NET\\_Framework\\_3.5#.NET\\_Framework\\_3.5](http://en.wikipedia.org/wiki/.NET_Framework_3.5#.NET_Framework_3.5)

## Izjava o samostojnosti dela

Izjavljam, da sem diplomsko delo izdelal samostojno pod vodstvom mentorja doc. dr. Marka Bajca, univ.dipl.ing. Izkazano pomoč drugih sodelavcev sem navedel v zahvali.

Dino Ivan Tot