

Univerza
v Ljubljani

Fakulteta
*za gradbeništvo
in geodezijo*



Jamova cesta 2
1000 Ljubljana, Slovenija
<http://www3.fgg.uni-lj.si/>

DRUGG – Digitalni repozitorij UL FGG
<http://drugg.fgg.uni-lj.si/>

To je izvirna različica zaključnega dela.

Prosimo, da se pri navajanju sklicujete na bibliografske podatke, kot je navedeno:

Pasarić, D., 2013. Izdelava napredne spletne karte za gorsko kolesarjenje. Diplomaska naloga. Ljubljana, Univerza v Ljubljani, Fakulteta za gradbeništvo in geodezijo. (mentor Petrovič, D.): 40 str.

University
of Ljubljana

Faculty of
*Civil and Geodetic
Engineering*



Jamova cesta 2
SI – 1000 Ljubljana, Slovenia
<http://www3.fgg.uni-lj.si/en/>

DRUGG – The Digital Repository
<http://drugg.fgg.uni-lj.si/>

This is original version of final thesis.

When citing, please refer to the publisher's bibliographic information as follows:

Pasarić, D., 2013. Izdelava napredne spletne karte za gorsko kolesarjenje. B.Sc. Thesis. Ljubljana, University of Ljubljana, Faculty of civil and geodetic engineering. (supervisor Petrovič, D.): 40 pp.

Univerza
v Ljubljani

Fakulteta za
*gradbeništvo in
geodezijo*



Jamova 2
1000 Ljubljana, Slovenija
telefon (01) 47 68 500
faks (01) 42 50 681
fgg@fgg.uni-lj.si

UNIVERZITETNI ŠTUDIJ
GEODEZIJE
SMER GEODEZIJA

Kandidat:

DARKO PASARIČ

**IZDELAVA NAPREDNE SPLETNE KARTE ZA
GORSKO KOLESARJENJE**

Diplomska naloga št.: 940/G

**CREATING ADVANCED WEB MAP FOR MOUNTAIN
BIKING**

Graduation thesis No.: 940/G

Mentor:

doc. dr. Dušan Petrovič

Predsednica komisije:

doc. dr. Alma Zavodnik Lamovšek

Somentor:

asist. dr. Klemen Kozmus Trajkovski

Član komisije:

doc. dr. Anka Lisec

asist. dr. Polona Pavlovčič Prešeren

Ljubljana, 28. 11. 2013

ERRATA

Stran z napako

Vrstica z napako

Namesto

Naj bo

IZJAVA O AVTORSTVU

Podpisani DARKO PASARIĆ izjavljam, da sem avtor diplomske naloge z naslovom:

»Izdelava napredne spletne karte za gorsko kolesarjenje«

Izjavljam, da je elektronska različica v vsem enaka tiskani različici.

Izjavljam, da dovoljujem objavo elektronske različice v repozitoriju UL FGG.

Ljubljana, 2013

Darko Pasarić

BIBLIOGRAFSKO – DOKUMENTACIJSKA STRAN IN IZVLEČEK

UDK:	528.94:796.61(043.2)
Avtor:	Darko Pasarić
Mentor:	doc. dr. Dušan Petrovič
Somentor:	asist. dr. Klemen Kozmus Trajkovski
Naslov:	Izdelava napredne spletne karte za gorsko kolesarjenje
Tip dokumenta:	Diplomska naloga – univerzitetni študij
Obseg in oprema:	40 str., 29 sl.
Ključne besede:	spletna karta, gorsko kolesarjenje, MTB, GPX datoteke, ASP.net, Google Maps API

Izvleček

V diplomski nalogi je predstavljena izdelava spletne karte, namenjene gorskim kolesarjem. Spletna karta sloni na Google-ovi aplikaciji Google maps. To pomeni, da uporabljamo Google-ove karte za prikaz poti in njenih oznak. Naloga večji del opisuje spletno programiranje in programski vmesnik Google Maps JavaScript API v3, ki nam omogoča integriranje interaktivne karte na spletno stran. Opisan je tudi označevalni jezik za izdelavo spletnih strani (HTML).

V nalogi so po vrsti obravnavana poglavja, ki opisujejo HTML, Google maps, že omenjeni programski vmesnik Google Maps API in uporabljene gradnike, nato je opisan postopek izdelave spletne karte, ki opisuje razvojna orodja, podatkovno bazo in spletne forme. Na koncu so dodana navodila uporabnikom, ki bodo dodajali GPS sledi na spletno stran.

BIBLIOGRAPHIC – DOCUMENTALISTIC INFORMATION AND ABSTRACT

UDK:	528.94:796.61(043.2)
Author:	Darko Pasarić
Supervisor:	Assist. Prof. Dušan Petrovič, Ph.D.
Cosupervisor:	Assist. Klemen Kozmus Trajkovski, Ph.D.
Title:	Creating advanced web map for mountain biking
Document Type:	Graduation Thesis – University studies
Notes:	40 p., 29 fig.
Key words:	web mapping, mountain biking, MTB, GPX files, ASP.net, Google Maps API

Abstract

The diploma presents the creation of a web map designed for mountain bikers. The web map is based on Google's application Google maps. This means that we use Google's maps to show the route and its markers. The thesis mostly describes web programming and the interface Google Maps JavaScript API v3 that enables us, to integrate the interactive map onto web page. It also describes the markup language for web pages (HTML).

In the thesis we discuss chapters such as HTML, Google maps, the before mentioned interface Google Maps API and used widgets, then we describe the process of creating the web map, development tools, database and web forms. Finally, we have written the instruction manual for those, who will add the GPS tracks on the web page.

ZAHVALA

Za pomoč in usmerjanje pri izdelavi diplomske naloge se zahvaljujem mentorju doc. dr. Dušanu Petroviču in somentorju asist. dr. Klemnu Kozmusu Trajkovskemu.

Zahvalil bi se tudi prijatelju Aleksandru Žagarju za pomoč pri programiranju spletne aplikacije in vsem, ki so me spodbujali pri pisanju diplome.

»Ta stran je namenoma prazna«

KAZALO VSEBINE

IZJAVA O AVTORSTVU	IV
BIBLIOGRAFSKO – DOKUMENTACIJSKA STRAN IN IZVLEČEK	V
BIBLIOGRAPHIC – DOCUMENTALISTIC INFORMATION AND ABSTRACT	VI
ZAHVALA	VII
1 UVOD.....	1
2 HTML	3
2.1 Pomembne HTML značke.....	4
3 GOOGLE MAPS.....	5
2.1 Karte in posnetki	5
2.1.1 Kartografska projekcija in koordinate	6
2.2 Google Maps JavaScript API v3	7
2.2.1 Osnovni prikaz karte na spletni strani.....	7
2.2.2 Dogodki.....	9
2.2.3 Kontrole	9
2.2.4 Prekrivni sloji	10
2.2.5 Tipi kart.....	11
4 IZDELAVA SPLETNE KARTE.....	14
4.1 ASP.NET in Microsoft Visual Studio 2010.....	14
4.2 Podatki in podatkovna baza.....	15
4.2.1 Kaj je GPX?	16
4.2.2 Microsoft SQL Server podatkovna baza.....	17
4.3 Struktura spletne strani.....	18
4.3.1 Spletna stran »Domov«	19
4.3.2 Spletna stran »Dodaj«.....	23
4.3.3 Spletna stran »Opis_dodatno«	26
4.3.4 Spletna stran »Opis«.....	27
5 NAVODILA IN POJASNILA UPORABNIKOM.....	32
5.1 Navodila za preproste uporabnike	32
5.2 Navodila za napredne uporabnike	34
6 ZAKLJUČEK.....	37
VIRI.....	39

KAZALO SLIK

Slika 1: Enostaven HTML dokument	3
Slika 2: Izvorna HTML koda.....	3
Slika 3: Tip karte ROADMAP	11
Slika 4: Tip karte SATELLITE	12
Slika 5: Tip karte HYBRID	12
Slika 6: Tip karte TERRAIN	13
Slika 7: Osnovno okno Microsoft Visual Studio (ASP.NET Web Application).....	15
Slika 8: Atributi podatkovne baze »poti«	17
Slika 9: Spletna stran z navigacijskim menijem in »Log in« kontrolo	18
Slika 10: Prikaz kontrol spletne forme Default.aspx	20
Slika 11: GridView kontrola z naslovnimi stolpci.....	21
Slika 12: Možne nastavitve pri GridView kontroli	22
Slika 13: SqlDataSource kontrola za povezavo z podatkovno bazo	22
Slika 14: Konfiguriranje podatkovne povezave.....	23
Slika 15: Spletna forma About.aspx	23
Slika 16: FileUpload kontrola	24
Slika 17: Pojavno okno »ListItem Collection Editor«	25
Slika 18: Spustni seznam za regijo (levo), za voznost (sredina) in za zahtevnost poti (desno)	25
Slika 19: Kontrola Button.....	26
Slika 20: Spletna forma Opis_dodatno.aspx	27
Slika 21: Spletna forma Opis.aspx	28
Slika 22: Graf višinske razlike	29
Slika 23: Pogovorno okno »Axis Collection Editor«.....	29
Slika 24: FormView kontrola v načinu urejanja predloge	30
Slika 25: ScriptManager kontrola.....	30
Slika 26: Domača stran naše spletne strani.....	32
Slika 27: Stran z opisom izbrane poti	33
Slika 28: Stran za dodajanje nove poti	34
Slika 29: Stran za ustvarjanje kažipota.....	36

OKRAJŠAVE IN SIMBOLI

HTML	Hyper Text Markup Language (Hipertekstovni označevalni jezik)
API	Application Programming Interface (Aplikacijski programski vmesnik)
GPS	Global Positioning System (Ameriški globalni navigacijski satelitski sistem za pozicioniranje)
CSS	Cascading Style Sheets (Kaskadne stilske podloge)
GPX	GPS eXchange format (GPS izmenjevalni format)
XML	Extensible Markup Language (Razširljiv označevalni jezik)
SQL	Structured Query Language (Strukturirani povpraševalni jezik)
UTC	Universal Time Coordinated (Univerzalni koordinirani čas)

»Ta stran je namenoma prazna«

1 UVOD

V štirih letih študija sem spoznal, da geodezija pokriva obsežno področje vedenja o merjenju in predstavitvi Zemlje predvsem iz tehničnega vidika. Z razvojem različnih tehnologij pa se le ta nekoliko razširi še na druga področja. Tukaj imamo v mislih računalništvo, ker večji del diplomske naloge sestoji iz programiranja spletne strani, vendar pa kljub vsemu ostaja končni izdelek geodetske narave, to je spletna karta. Prikaz Zemeljskega površja na karti pa umestimo v področje geodezije, natančneje kartografije in topografije.

Predstavljajmo si, da se odpravimo na kolesarsko turo po gozdu, katere poti še ne poznamo. Prišli smo na osvojeni vrh kateremu sledi spust nazaj v dolino. Na voljo imamo več poti, vendar ne vemo, kam katera pelje. Smo izčrpani in v časovni stiski, torej je vprašanje, kako varno in pravočasno priti nazaj domov. Tako smo v naši diplomski nalogi poskusili rešiti ta problem z izdelavo spletne karte gorskih kolesarskih poti, ki bo preprečevala takšne in podobne situacije. S tem bomo omogočili uporabnikom oz. kolesarjem, da si predhodno ogledajo pot, na katero se želijo podati, in pridobijo o njej čim več uporabnih informacij. Tako bodo točno vedeli, kam zaviti, koliko časa potrebujejo za pot in kako dolga ter zahtevna je le ta, ter se s tem izognili neprijetnim dogodivščinam.

Zato smo se lotili izdelave napredne spletne karte, ki bo vsebovala čim več informacij na enem mestu. Zakaj na enem mestu? Ker obstaja že kar nekaj spletnih strani s podobno tematiko in vsebino (tudi pri nas v Sloveniji), vendar je ta razkropljena po več straneh: tako imamo opisne podatke o poti na lastni strani, grafični prikaz poti na karti pa moramo pogledati nekje pri drugih ponudnikih spletnih kart. Zato smo se odločili, da to dvoje združimo in hkrati omogočimo še uporabnikov prispevek k polnjenju podatkovne baze z novimi gorsko kolesarskimi potmi.

Ko uporabnik bere opis poti dobi kup informacij, ki bi jih lahko razdelili na dva dela. Ene predstavljajo in opisujejo doživetje poti ter podroben opis, med tem ko so druge bolj specifične narave, npr. kam in kje je treba zaviti, kakšna je zahtevnost dela proge, kakšne so razmere itd. Torej pridemo do kupa informacij, ki si jih je težko vse zapomniti. Zato smo v nalogi ta problem skušali rešiti z interaktivnim kažipotom.

1.1 Pregled vsebine

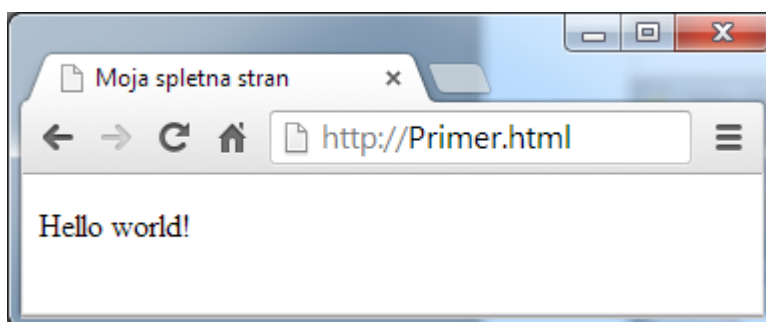
Omenili smo že, da naša naloga sloni na spletnem programiranju, ki nam omogoča, da lahko izdelamo napredno spletno karto. Za začetek imamo kratko pojasnjeno, kako se napiše oz. ustvari neka enostavna spletna stran (HTML). Nato se lotimo preučevanja Google-ovih kart, kratkega opisa zgodovine, kakšne vrste podlag nam omogočajo, od kje so posnetki pridobljeni, kdaj so posneti in v kateri kartografski projekciji so le ti. Ker govorimo o spletni karti na naši spletni strani in ne o splošni karti, ki jo Google ponuja na svojih spletnih straneh, smo uporabili programski vmesnik Google maps API, ki omogoča integracijo Google-ove karte na uporabnikovo spletno stran. Opisali smo programski

vmesnik, kako ga uporabimo in predstavljениh imamo peščico gradnikov, ki nam omogočajo opremljanje oz. oplemenitenje naše karte. V teh poglavjih je povzetek teorije, ki jo v obsežnejši obliki najdemo tudi na spletu, naslednja poglavja pa opisujejo potek izdelave spletne karte in uporabo le te. Predstavljena so razvojna orodja, struktura spletne strani in posamezne spletne forme - predloge, ki se pošljejo strežniku, da nam prikaže spletno stran. Da bodo uporabniki vedeli, kako stran deluje in kako se doda novo posneto pot na spletno stran, smo napisali tudi navodila za uporabo.

2 HTML

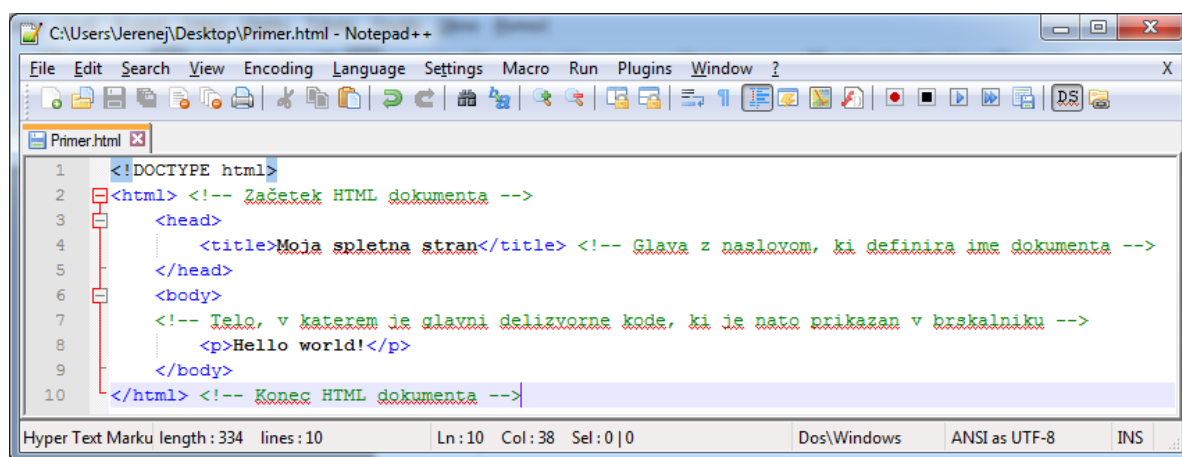
Za lažje razumevanje spletnega programiranja in delovanja spletne strani, bomo za začetek pojasnili osnovo spletnega programiranja, to je HTML, brez katerega si spletnih strani ne znamo predstavljati.

Hyper Text Markup Language (Hipertekstovni označevalni jezik) je jezik, ki se uporablja za izdelavo spletnih strani. Njegovi začetki segajo v leto 1990, od takrat je doživel že kar nekaj sprememb. Ker je izvorna koda HTML-ja običajna tekstovna datoteka in za izvajanje ne potrebuje prevajalnika, odpremo jo kar v brskalniku (slika 1), jo je moč napisati praktično v vsakem tekstovnem urejevalniku (Beležnica, Notepad++, ipd.).



Slika 1: Enostaven HTML dokument

Izvorna koda (slika 2) je sestavljena iz tako imenovanih značk (angl. tag) in jo lahko razdelimo na dva glavna dela, glavo (angl. head) in telo (angl. body). Kljub temu, da je koda standardizirana, je lahko prikaz strani v različnih brskalnikih različen.



Slika 2: Izvorna HTML koda

Ker pa starejše različice HTML-ja ne omogočajo izdelave dinamičnih spletnih strani (dinamični gradniki, animacije, ipd.), se pogosto uporablja v povezavi z drugimi programskimi jeziki in orodji, npr.: Flash, JavaScript, PHP, s čimer močno razširimo njegovo uporabnost.

2.1 Pomembne HTML značke

V nadaljevanju bomo predstavili nekaj najpogosteje uporabljenih HTML značk. Običajno za sestavo enega elementa (npr. naslova) uporabimo dve znački, in sicer značko za začetek `<h1>` in značko za konec elementa `</h1>` (poševnica pove, da se element, v tem primeru naslov h1, zaključuje), med njima pa se nahaja besedilo. V začetno značko lahko, če le ta to tudi omogoča, dodamo še attribute za barvo, velikost, poravnavo (`<h1 align=center>`, v tem primeru povemo, da bomo naslov h1 poravnali na sredino), idr. Pa si pogledjmo najpogosteje uporabljene značke:

- `head`, v glavi HTML dokumenta so zapisani podatki o naslovu/imenu strani ter razni meta podatki,
- `body`, znotraj tega elementa se nahaja celotna koda, ki bo prikazana v brskalniku,
- `br`, označuje odstavek,
- `div`, združi več elementov v enega, pogosto se uporablja v kombinaciji z različnimi atributi, ki veljajo za celotno skupino (odebeljen tisk, font, velikost ter barva pisave npr.: `<div style="color: #0900C4; font: Helvetica 12pt;border: 1px solid black;"> ostala koda </div>`
- `h1` do `h6`, značke označujejo naslove, pri čemer je `h1` glavni naslov, `h6` pa zadnji podnaslov,
- `img`, značka omogoča vstavljanje slike na spletno stran,
- `a`, značka omogoči vključitev povezave na drugo spletno stran znotraj dokumenta ali pa v svetovnem spletu (primer povezave do iskalnika GOOGLE: ` Google `)

Zgoraj naštetih značk so samo osnovne, z njimi lahko izdelamo osnovno spletno stran, obstaja pa še kup drugih značk za izdelavo npr. različnih tabel, okvirjev, s katerimi razdelimo spletno stran na dele, in mnoge druge (Kaltenekar, 2006).

3 GOOGLE MAPS

Ko brskamo po spletu, posebej ko iščemo informacije o lokacijah, pogosto naletimo na spletno stran z opisno vsebino in pa okencem s karto. Takšno okence je lahko izdelek programskega vmesnika Google maps API. Google namreč ni edini ponudnik tovrstnih storitev, saj na trgu obstajajo tudi druga podjetja in razvijalci z več ali manj podobnimi storitvami.

API (Application Programming Interface) aplikacijski programski vmesnik določa, kako naj bi programske komponente vplivale ena na drugo. V objektno orientiranih jezikih, kot je JavaScript, API običajno vključuje določitev razredov in njihove metode (API, 2013).

Google maps lahko definiramo kot prosto dostopen strežnik s kartami in geografskimi podatki. Google ga je pridobil s prevzemom podjetja Where 2 Technologies bratov Larsa in Jensa Rasmussen leta 2004, kjer je šlo sprva za c++ program, ki so ga lahko uporabniki prenesli na svoj računalnik. Kmalu se je ta program sprevrgel v hitro rastočo spletno aplikacijo, ki je bila prvič objavljena na Googlovem blogu 8. februarja 2005 in je delovala samo na spletnih brskalnikih Internet Explorer in pa Mozilla. Kmalu za tem, junija 2005, je Google ponudil na trg najbolj obsežen in tudi najpogosteje uporabljen API za integracijo njihovih kart v druge spletne strani in aplikacije (History, 2013). Torej nam Google maps API omogoča vključitev kart v lastne spletne strani in aplikacije, bodisi razvijalcev bodisi uporabnikov.

Spletna aplikacija Google Maps je skoraj v celoti napisana v programskem jeziku JavaScript in XML. Mi bomo prav tako pisali v JavaScript-u, saj ta omogoča ustvarjanje dinamičnih spletnih strani (ali karte), ki za razliko od običajne statične karte v primeru prikaza velikega števila informacij ne postane neberljiva in nepregledna (Zaversnik, 2013). Torej, omogočeno nam je posamezno prikazovanje v danem trenutku zanimivih informacij na karti. Da vse to poteka brez osveževanja spletne strani, pa poskrbi AJAX, tj. skupina medsebojno povezanih spletnih razvojnih tehnik, ki se uporabljajo za ustvarjanje interaktivnih spletnih aplikacij. Aplikacije, narejene z Ajax-om, si v ozadju s strežnikom asinhrono izmenjujejo samo podatke, ki jih dejansko potrebujejo, zato vse poteka hitro in tekoče brez ponovnega nalaganja strani (Ajax, 2013). Če povzamemo, JavaScript poskrbi za dinamično izvajanje spletne strani, Ajax pa za hitro in tekoče delovanje strani brez osveževanja. Karta, ki jo vidimo na ekranu, je sestavljen iz več manjših slik oz. posnetkov, ki so naloženi v ozadju.

2.1 Karte in posnetki

Google maps ponuja bolj ali manj podroben prikaz zemeljskega površja iz letalskih in satelitskih posnetkov. Google posplošeno uporablja besedo »satellite«, kar pa strogo gledano ne drži v celoti, saj so visokoločljivostne (angl. high-resolution) podobe dejansko letalski posnetki, posneti iz letala, ki leti na višini od 150 do 450 m. Predvidevamo, da se letalski posnetki večinoma uporabljajo na urbanih

področjih ter na področjih naravnih in kulturnih znamenitosti, medtem ko ostali del površja pokrivajo satelitski posnetki.

Slabost uporabljenih posnetkov je njihova zastaranost. Namreč posnetki so lahko stari 5 ali več let, med tem časom pa se je lahko že marsikaj spremenilo, predvsem spremembe razne infrastrukture in pozidave novih objektov, ki so posledica hitrega razvoja ali pa celo kakšne naravne katastrofe.

Za običajne uporabnike spleta je Google maps zelo uporaben, posebej zaradi visokoločljivostnih posnetkov, medtem ko so se nekatere države oz. njihove vlade pritoževale, češ da je to dober vir za načrtovanje terorističnih napadov. Ta očitok je Google rešil tako, da je zameglil nekatere objekte visokega pomena za državno varnost, predvsem v ZDA (Security concerns, 2013).

Podatke, ki jih Google Maps uporablja v svojih storitvah, je v začetku pridobival od ponudnika Navteq, leta 2008 pa je v celoti prešel na TeleAtlas. Danes oz. od leta 2011 naprej Google uporablja kar se da več svojih lastnih podatkov in podatke lokalnih agencij in inštitucij za kartografijo in geodezijo posameznih držav (Schwartz, 2011).

2.1.1 Kartografska projekcija in koordinate

Google Maps karte temeljijo na Mercatorjevi projekciji. Zemljo preslikamo na karto kot kroglo, posledično pride do rahlega odstopanja koordinat pri manjših območjih zaradi neskladja med WGS 84 elipsoidom in Mercatorjevo sferično projekcijo. Mercatorjeva projekcija je konformna cilindrična projekcija, kjer so vzporedniki in poldnevnik ravne linije, med seboj pravokotne, koti pa se ohranjajo. Loksodroma, črta, ki tvori konstantne kote z meridiani se preslika kot ravna linija, kar je poleg konformnosti ena pomembnejših lastnosti Mercatorjeve projekcije. Slabost te projekcije je, da se površine z oddaljevanjem od ekvatorja čedalje bolj deformirajo in ne prikazujejo pravilne velikosti ploskovnih in linijskih objektov, polov pa v celoti sploh ni mogoče prikazati, saj je linearno merilo na polih neskončno. Zaradi tega se v projekciji svet prikazuje samo v območju do geografske širine 85° (Peterca in sod., 2001). Ta projekcija je najbolj uporabna za navigacijo, medtem ko za Google Maps karte, kljub povečevanju deformacije z oddaljevanjem od ekvatorja, postane uporabna, ker gre za interaktivno karto, ki omogoča približevanje izbranega področja, kjer prihaja do relativno manjših deformacij (Mercator projection, 2013).

Ločiti moramo med koordinatami na Zemlji, za katere Google uporablja datum WGS84, in pa med koordinatami na karti v Mercatorjevi projekciji, katere je treba za vsako točko, ki jo želimo prikazati na karti, preračunati.

2.2 Google Maps JavaScript API v3

Google Maps API je programski vmesnik, ki omogoča vstavljanje Google-ovih kart v lastne spletne strani. Sestavljen je iz datotek JavaScript, ki vsebujejo razrede z metodami in lastnostmi, ki jih uporabljamo, da povemo karti, kaj naj prikaže.

API ponuja številne koristne možnosti, s katerimi lahko našo karto poljubno opremimo in dodamo razne funkcionalnosti. Za vsebinsko oplemenitev karte poskrbijo razni servisi, kot so Directions service, Distance Matrix, Elevation, Geocoding, Maximum Zoom Imagery in Street View Service.

Da lahko začnemo uporabljati Google Maps API, moramo najprej pridobiti tako imenovani API Key (ključ). Tako lahko nadzorujemo porabo uporabljenih map oz. kart in zagotovimo Googlu dostop do naše aplikacije, če bi bilo to potrebno. Za pridobitev ključa je potrebno ustvariti račun pri Googlu.

Google Maps API je prosto dostopen za komercialno uporabo, vendar pa obstaja omejitev porabe Google-ovih kart z uporabo Google Maps JavaScript API v3, ki je omejen na 25.000 naloženih kart na dan. Nalaganje karte se šteje takrat, ko je le ta naložena na spletno stran, tako da se lahko poljubno premikamo po karti ali približujemo/oddaljujemo sliko, ne da bi vplivali na porabo prenesenih podatkov. Če presežemo omejitev porabe, nam Google ponuja tri možnosti in sicer, da zakupimo dodatno količino podatkov, kupimo licenco za podjetja ali pa spremenimo svojo stran tako, da bo poraba pod mejo omejitve.

Dokumentacija je zasnovana tako, da jo najbolje razumejo ljudje, ki se ukvarjajo s programiranjem v JavaScript-u in z objektno orientiranimi programskimi koncepti. Iz uporabniškega stališča pa moramo vsaj malo poznati samo Google Maps aplikacijo (Google – Google Maps API, 2013).

2.2.1 Osnovni prikaz karte na spletni strani

Da lahko začnemo uporabljati Google Maps API, moramo v naš HTML dokument napisati naslednji odstavek.

```
<script  
src="https://maps.googleapis.com/maps/api/js?key=API_KEY&sensor=SET_TO_TRUE_OR_FALSE"  
type="text/javascript"></script>
```

URL v zgornjem stavku je lokacija JavaScript datoteke, ki naloži vse simbole in definicije, ki jih potrebujemo, da lahko uporabljamo Google Maps API. Ta script stavek je obvezen.

Tukaj je za nas najbolj pomemben parameter `key`, kjer moramo napisati svoj API ključ, ki smo ga predhodno pridobili. Izbrati moramo tudi med vrednostima `TRUE` ali `FALSE` parametra `sensor`, s katerim določimo, ali naša aplikacija uporablja GPS sprejemnik ali ne.

Z naslednjim korakom rezerviramo prostor za karto na naši spletni strani. Po navadi to naredimo z `div` elementom. S tem definiramo, kateri element se bo prikazal v brskalniku in kako. V našem primeru je to element z identifikacijo `map-canvas`. Temu elementu smo določili tudi višino in širino okna, v katerem bo prikazana karta, to smo storili s kontrolo `style`.

```
<div id="map-canvas" style="width: 100%; height: 100%"></div>
```

Kako bo naša osnovna karta prikazana, določimo z nastavitvami, kot so `center`, kjer definiramo sredino karte s parametroma zemljepisne širine in dolžine, `zoom` določa povečavo oz. pomanjšavo karte, `mapTypeId` pa določi, ali gre za normalno 2D karto, satelitsko sliko, prikaz terena ali pa mešanico dveh ali več slojev.

```
var mapOptions = {  
  center: new google.maps.LatLng(-34.397, 150.644),  
  zoom: 8,  
  mapTypeId: google.maps.MapTypeId.ROADMAP  
};
```

Spodnji del kode izriše karto na naši spletni strani, upoštevajoč zgoraj določen prostor in nastavitve osnovne karte.

```
var map = new google.maps.Map(document.getElementById("map-  
canvas"),  
  mapOptions);
```

Kako in kdaj se bo karta izrisala, določimo z dogodkom `onload`, ki pokliče našo funkcijo `initialize()`. S tem se izognemo morebitnim težavam pri nalaganju strani in posledično tudi naše karte.

```
<body onload="initialize()">
```

2.2.2 Dogodki

JavaScript programski jezik v brskalniku deluje na podlagi dogodkov, kar pomeni, da se odziva na interakcijo in prav na tak način deluje tudi Google Maps API. Kot dogodke lahko razumemo npr. klik z miško ali pa spreminjanje nivoja povečave.

Z naslednjo kodo podamo t.i. poslušalca dogodkov, kjer namesto objekta `marker` podamo poljuben objekt, nad katerim želimo spremljati dogodke. Drugače povedano, spodnja koda ob kliku na oznako sproži funkcijo, ki spremeni nivo povečave (angl. `zoom`) na 8.

```
google.maps.event.addListener(marker, 'click', function() {  
    map.setZoom(8);  
});
```

2.2.3 Kontrole

Karta, ki jo prikažemo na naši spletni strani preko vmesnika Google Maps API, vsebuje uporabniške elemente za interakcijo karte. To so osnovne kontrole, ki omogočajo premikanje po karti, nivo povečave in izbiro podlage karte. Te tri kontrole se vedno prikažejo na karti brez kakršnih koli nastavitev ali dodatnega pisanja kode. Torej so prednastavljene, ampak jih lahko izklopimo z ukazom `disableDefaultUI: true`, ki ga napišemo v funkciji `initialize()`.

Dodajanje oz. odstranjevanje kontrol na/iz karte izvedemo tako, da napišemo ime kontrole, in pa ali je resnična (`true`) ali ne (`false`), kar pomeni, da so te spremenljivke tipa `boolean`.

```
{  
    panControl: boolean,  
    zoomControl: boolean,  
    mapTypeControl: boolean,  
    scaleControl: boolean,  
    streetViewControl: boolean,  
    overviewMapControl: boolean  
}
```

Zgoraj so po vrsti našete kontrole za premikanje po karti, za povečavo, tip karte, merilo karte, kontrola za pogled iz ulice in kontrola za preglednost karte.

2.2.4 Prekrivni sloji

Za naše diplomsko delo so precej pomembni gradniki, saj nam le ti omogočajo izris poti in pripadajoče opreme na karti. Ti sloji so gradniki na karti, nanašajoč se na horizontalne koordinate, tako da se lahko premikajo skupaj z mapo, kadar se po njej premikamo. Z njimi označujemo in rišemo objekte, ki so za nas pomembni, in jih želimo prikazati ali izpostaviti. Če izberemo nekaj pogosto uporabljenih gradnikov, so to oznake, linije, poligoni, slike, informacijska okna in pa lastni sloji. Nekaj jih bomo na kratko opisali v nadaljevanju.

MARKER – Markerje oz. oznake uporabljamo za označevanje točkovnih elementov na karti, in sicer s standardno ikono, ki jo lahko prilagodimo svojemu okusu. Pri konstruiranju oznake sta še posebej pomembna dva parametra, in to sta lokacija (`position`) s podano zemljepisno širino in dolžino ter karta (`map`), na kateri se bo oznaka izrisala. Oznake so lahko tudi interaktivne, saj se odzivajo na klik z miškinim gumbom, kar pomeni, da jih je možno animirati, premikati, itd.

Spodnji primer izriše enostavno oznako na lokaciji `myLatLng`.

```
var myLatLng = new google.maps.LatLng(46.240000, 15.270000);
var marker = new google.maps.Marker({
  position: myLatLng,
  map: map
});
```

POLYLINE – Lomljenko uporabimo, kadar želimo prikazati linijske objekte. Narišemo lahko posamezno črto ali pa linije povežemo med seboj. Polyline gradnik sestoji iz seznama zemljepisnih širin in dolžin posameznih točk. Za več posameznih črt potrebujemo več Polyline objektov. Nastavitve, ki jih lahko spreminjamo določeni liniji, so barva, prosojnost in debelina črte.

Spodaj je primer treh medsebojno povezanih linij.

```
var flightPlanCoordinates = [
  new google.maps.LatLng(37.772323, -122.214897),
  new google.maps.LatLng(21.291982, -157.821856),
  new google.maps.LatLng(-18.142599, 178.431),
  new google.maps.LatLng(-27.46758, 153.027892)
];
var flightPath = new google.maps.Polyline({
  path: flightPlanCoordinates,
  strokeColor: "#FF0000",
  strokeOpacity: 1.0,
  strokeWeight: 2
});
```

POLYGON – Poligoni so podobni linijskim objektom, za razliko od tega, da so zaključene celote, torej imata prva in zadnja točka iste koordinate. Poleg zgoraj omenjenih nastavitev pri linijah velja tukaj isto, s tem da določimo tudi barvo in prosojnost območja znotraj zaključenega poligona.

```
poligon = new google.maps.Polygon
```

2.2.5 Tipi kart

Izbiramo lahko med štirimi osnovnimi tipi kart:

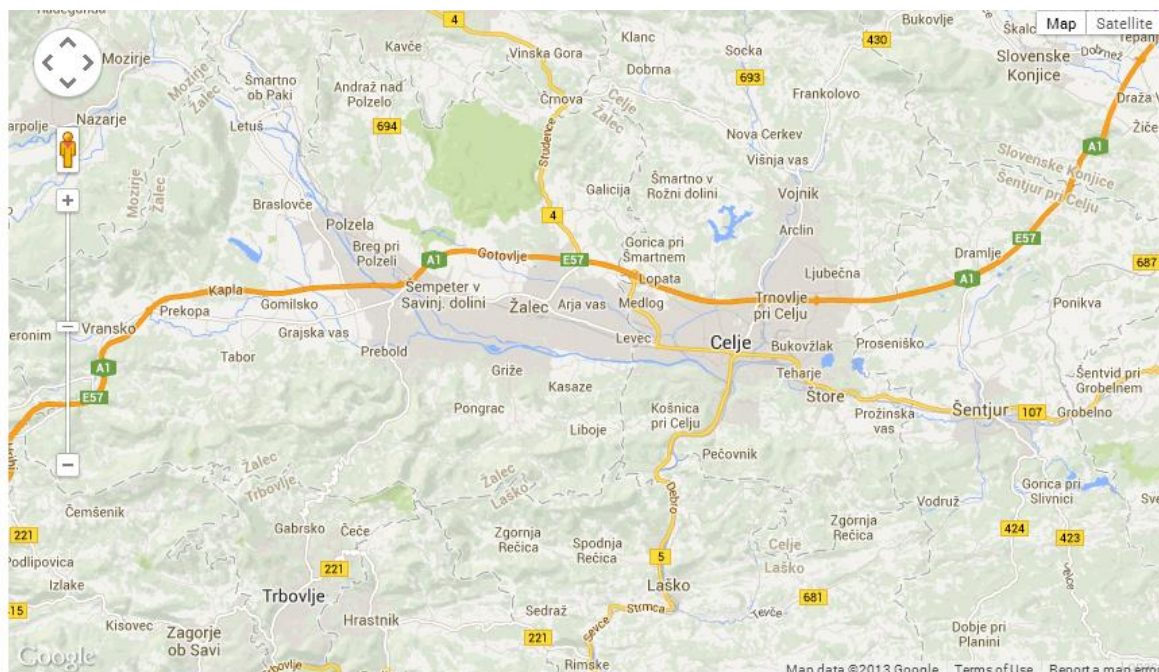
ROADMAP – je avtokarta (slika 3)

SATELLITE – je satelitska slika (slika 4)

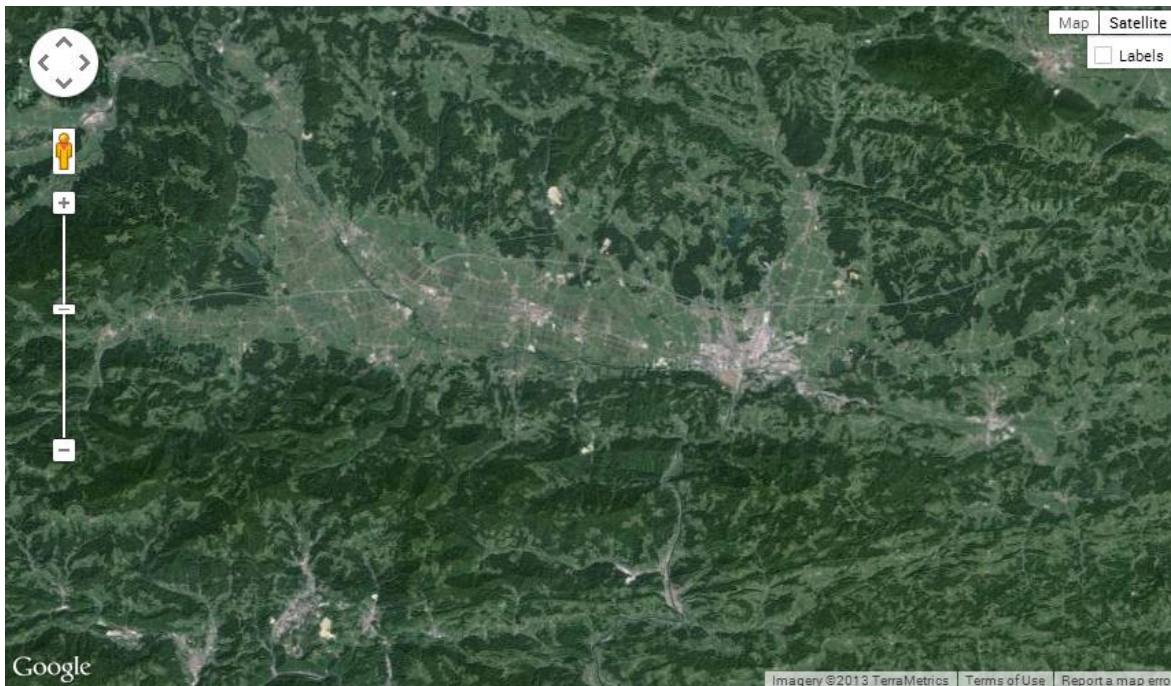
HYBRID – je satelitska slika z izbranimi sloji avtokarte (ceste, zemljepisna imena,...) (slika 5)

TERRAIN – je senčen prikaz površja s terenskimi značilnostmi (relief, plastnice) z izbranimi sloji avtokarte (slika 6)

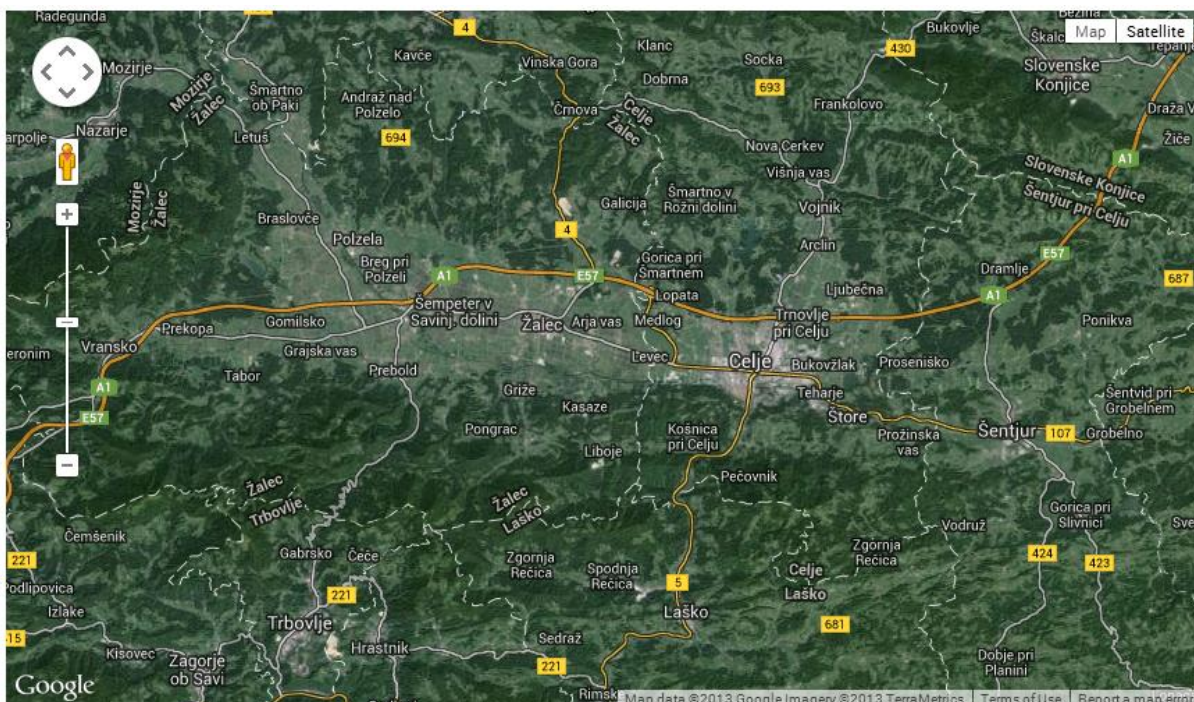
Zahtevnejši uporabniki pa imajo možnost prilagajanja in spreminjanja teh obstoječih kart.



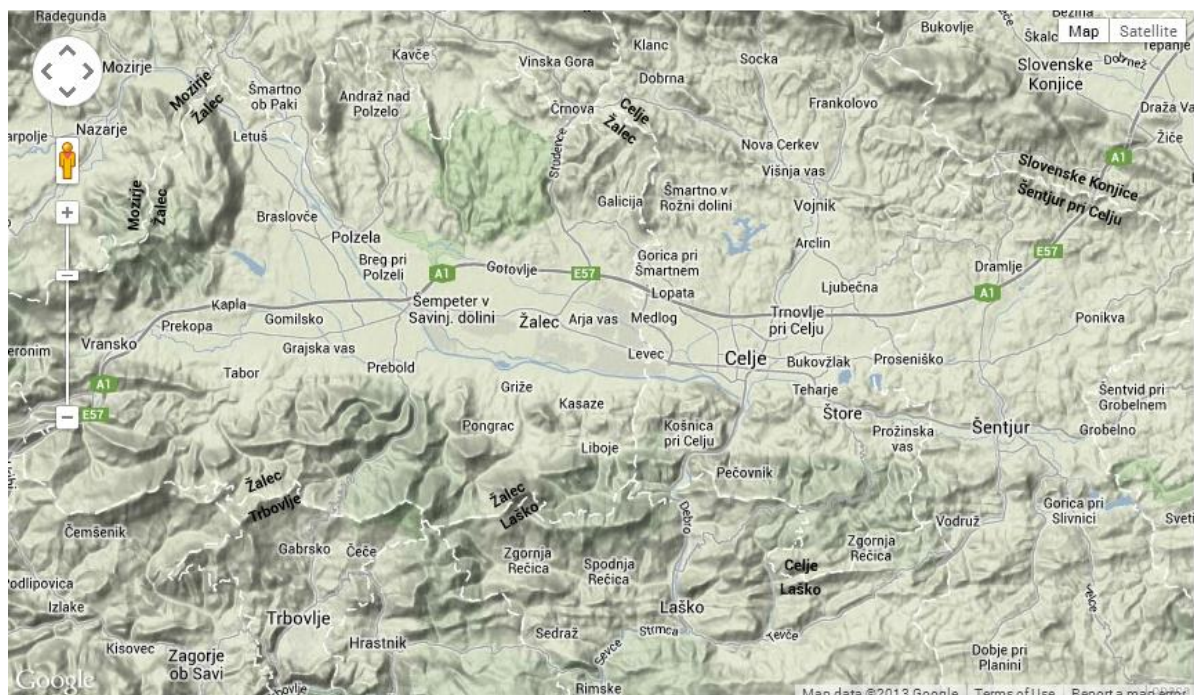
Slika 3: Tip karte ROADMAP



Slika 4: Tip karte SATELLITE



Slika 5: Tip karte HYBRID



Slika 6: Tip karte TERRAIN

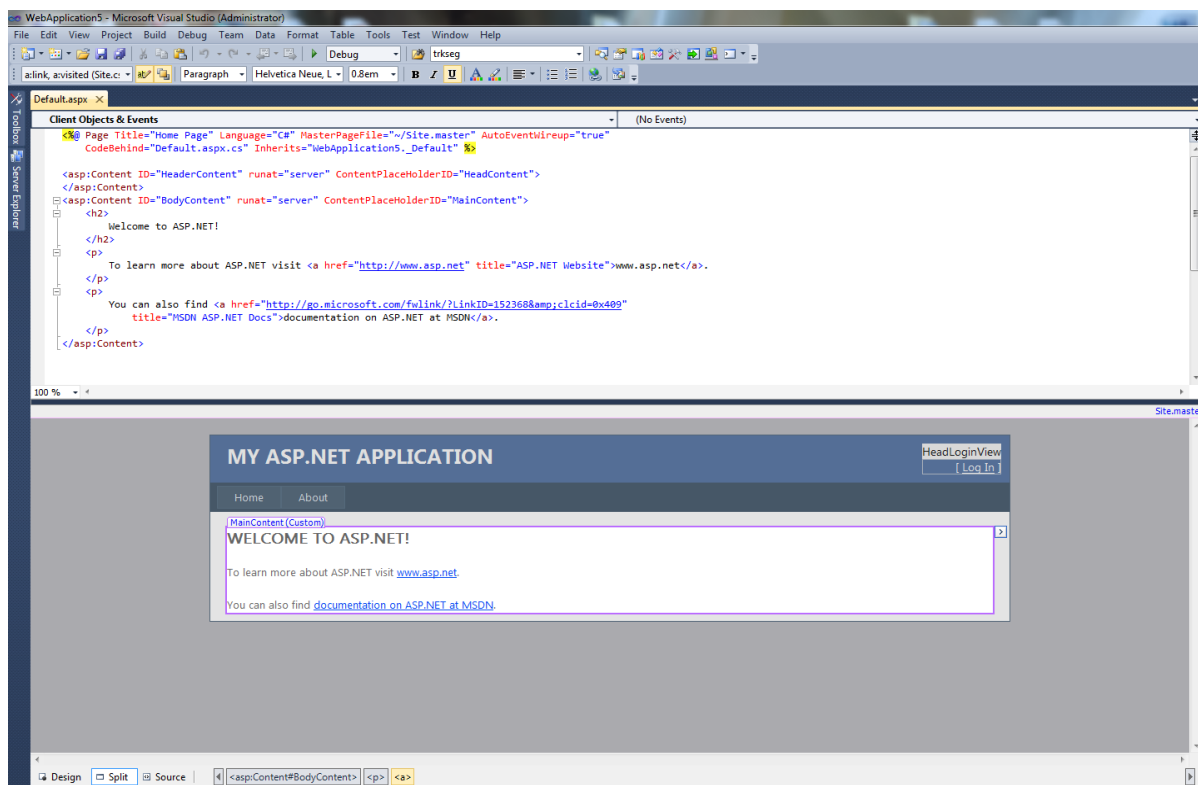
4 IZDELAVA SPLETNE KARTE

To poglavje je namenjeno predstavitvi izdelave spletne karte za gorske kolesarje. Sama naloga zahteva predvsem znanje spletnega programiranja, zato je bilo potrebno dodatno individualno učenje določenih programskih jezikov in okolij. Potrebna orodja za izdelavo spletne karte, ki smo jih potrebovali za uspešno izvedeno nalogo, so ASP.NET, Microsoft Visual Studio 2010, Microsoft SQL Server 2008 in pa Google-ov spletni servis oz. programski vmesnik Google Maps JavaScript API v3, kateri nam omogoča, da prikažemo interaktivno karto na svoji spletni strani.

V naslednjih poglavjih je napisanega nekaj več o orodjih ASP.NET in Microsoft Visual Studio, pojasnili bomo, od kje in kako pridobimo vhodne podatke, kako jih shranimo v podatkovno bazo in kako jih kasneje predstavimo. Predstavili bomo tudi strukturo spletne strani in njene forme.

4.1 ASP.NET in Microsoft Visual Studio 2010

Microsoft Visual Studio smo uporabili kot razvojno okolje za izdelavo spletnih strani in spletnih aplikacij. Microsoft Visual Studio podpira več različnih programskih jezikov, med katerimi smo izbrali C#. Podpira tudi jezike, kot so HTML/XHTML, CSS in JavaScript (Bonačič, 2008). Če našete jezike nekako opredelimo, bi lahko rekli, da s HTML-jem prikažemo spletno stran v brskalniku, CSS poskrbi za izgled in obliko spletne strani, C# in JavaScript pa določata delovanje in izvajanje določenih funkcij spletne strani. Kar omogoča enostavnejšo izdelavo dinamičnih spletnih strani, pa je dejansko ASP.NET, ki je spletno aplikacijsko ogrodje (angl. Web application framework). To ogrodje ponuja že spisane kontrole oz. gradnike, ki jih samo dodajamo na neko predlogo oz. šablono (angl. template) naše spletne strani, ter jim povemo, kaj naj vsebujejo, kako naj funkcionirajo ali se odzivajo (Glenn, 2003).



Slika 7: Osnovno okno Microsoft Visual Studio (ASP.NET Web Application)

Na sliki 7 je prikazana osnovna stran spletne aplikacije, na kateri gradimo in spreminjamo že obstoječe gradnike, da dosežemo svoj cilj oz. ustvarimo končen izdelek. To je skupek vseh zgoraj omenjenih orodij, ki omogočajo ustvarjanje naše spletne strani. Podrobneje gledano imamo prikazano spletno formo (angl. Web form) z razdeljenim pogledom na razvojno ogrodje ASP.NET. Zgornji del se imenuje izvorna koda (angl. Source), s katero ustvarjamo spletno stran zgolj s tipkanjem po tipkovnici in pri kateri moramo pisati tudi HTML značke in vse, kar spada zraven. Spodnji del, tako imenovani načrtni del (angl. Design), pa je dosti enostavnejši. Nanj samo povlečemo poljubno kontrolno iz seznama orodij (angl. Toolbox) ter ji po potrebi spreminjamo nastavitve (angl. Properties) in dodajamo vsebino. Medtem pa se v izvorni kodi dogajajo »čudeži«, saj se le ta sama sprti izpisuje (Microsoft – ASP.NET, 2013).

4.2 Podatki in podatkovna baza

Da lahko komu ponudimo uporabne informacije, potrebujemo za to odgovarjajoče podatke, saj lahko le uporaben in ovrednoten podatek postane informacija. V našem primeru se želimo informirati o gorsko kolesarski poti, po kateri se imamo namen peljati.

Glavni vir naših podatkov je *.gpx datoteka.

4.2.1 Kaj je GPX?

GPX (GPS eXchange format) je XML podatkovni format za izmenjavo podatkov GPS (vmesne točke, poti, sledi) med različnimi aplikacijami, programi in med uporabniki (Topografix, 2013). Drugače povedano, datoteke GPX lahko razumejo oz. uporabljajo tudi drugi programi in ne samo tisti, s katerimi so bile ustvarjene.

Pojavil se je nov izraz XML (Extensible Markup Language), to je razširljiv označevalni jezik, ki ga pogosto srečamo pri brskanju po internetu. Je jezik, ki omogoča format za opisovanje strukturiranih podatkov ter s tem olajša izmenjavo in prenos podatkov med več omrežji (XML, 2013).

Podrobnejši vpogled v datoteko GPX razkrije sledečo strukturo podatkov, ki jo bomo prikazali kar na konkretnem primeru.

```
<?xml version="1.0" encoding="UTF-8"?>
<gpx
  version="1.0"
  creator="GPSBabel - http://www.gpsbabel.org"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://www.topografix.com/GPX/1/0"
  xsi:schemaLocation="http://www.topografix.com/GPX/1/0"
<wpt lat="46.233944" lon="15.246547">
<ele>273.536530</ele>
<time>2012-11-11T11:40:10Z</time>
<name>Lat:46.233944, Lon:15.246547, Ele:273.536530</name>
</wpt>
<trk>
  <name>4-1T</name>
<trkseg>
<trkpt lat="46.510916667" lon="15.082316667">
  <ele>408.000000</ele>
  <time>2007-12-31T09:54:10Z</time>
</trkpt>
<trkpt lat="46.510416667" lon="15.082350000">
  <ele>409.000000</ele>
  <time>2007-12-31T09:55:32Z</time>
</trkpt>
</trkseg>
</trk>
</gpx>
```

Zgornji primer je zelo skrajšana verzija običajne GPX datoteke, saj le ta po navadi vsebuje veliko več točk. Ampak za razumevanje strukture podatkov je to več kot dovolj.

Tako lahko vidimo, da vsebuje predstavljeni primer podatke o verziji XML-ja in GPX-a, ime proizvajalca naprave, s katero je bila ustvarjena, in pa unikatno ime (»namespace«), na katerega se nanašajo elementi in atributi. Uporabni podatki, ki jih vsebuje vsaka točka »trkpt« ali vmesna točka »wpt«, so zemljepisna širina in dolžina (»lat« in »lon«), višina (»ele«) ter čas (»time«). Vsi naštetih atributi so zapisani ali znotraj značke <wpt> ali <trkpt> ali <ele> ali značke <time>. Lahko pa

datoteka vsebuje več elementov za dodatni opis, kot so ime točke, hitrost, url naslov, geoidna višina itd.

Geografske koordinate so zapisane v decimalnih stopinjah, nadmorska višina je podana v metrih, čas pa v UTC (angl. Universal Time Coordinated) formatu. UTC je univerzalni koordinirani čas, ki ga dobimo, če našemu srednjeevropskemu času odštejemo 1 uro v zimskem času in 2 uri v poletnem času.

4.2.2 Microsoft SQL Server podatkovna baza

Podatkovna baza je organizirana zbirka med seboj pomensko povezanih podatkov, do katerih dostopamo z ustvarjeno podatkovno povezavo. Za upravljanje naše baze smo uporabili Microsoft SQL Server 2008, ki je sistem za upravljanje z relacijskimi podatkovnimi bazami (Microsoft SQL server, 2013).

Ustvarili smo podatkovno bazo »poti«, ki vsebuje attribute, prikazane na sliki 8. Vsakemu atributu določimo podatkovni tip, ali bo to število, znakovni niz, čas ali pa enolični identifikator. Podatkovni tipi, ki smo jih uporabili, so:

- *int* celo število, *real* pa realno število,
- *varchar(50)*, *nvarchar* in *text* so znakovni nizi, ki se med seboj razlikujejo po dolžini in po standardu za kodiranje znakov v računalništvu,
- *time(7)*, je zapis 24 urnega časa z natančnostjo na 7 decimalk sekunde (*hh:mm:ss.nnnnnnn*),
- *uniqueidentifier*, je unikatna vrednost, sestavljena iz šestnajstiškega številskega sistema.

	Column Name	Data Type	Allow Nulls
▶	id	int	<input type="checkbox"/>
	imePoti	varchar(50)	<input checked="" type="checkbox"/>
	regija	varchar(50)	<input type="checkbox"/>
	dolzina	real	<input checked="" type="checkbox"/>
	cas	time(7)	<input checked="" type="checkbox"/>
	zahtevnost	varchar(50)	<input type="checkbox"/>
	opis	text	<input checked="" type="checkbox"/>
	voznost	varchar(50)	<input checked="" type="checkbox"/>
	gpx_file_path	nvarchar(MAX)	<input checked="" type="checkbox"/>
	added_by_user	uniqueidentifier	<input type="checkbox"/>
			<input type="checkbox"/>

Slika 8: Atributi podatkovne baze »poti«

V podatkovno bazo se zapišejo podatki, ki jih vnesemo ročno ali pa se izračunajo na podlagi podatkov iz *.gpx datoteke.

- **id** je primarni ključ, ki enolično določa vsak zapis, da ne pride do podvajanja zapisov, in se samodejno zapisuje v bazo.
- **imePoti** je polje, v katerega shranimo ime poti, zapisanega v enovrstično tekstovno okno.
- **regija** je polje, v katerega se zapiše izbrana vrednost iz spustnega seznama za regije.
- **dolzina** je polje, v katerega se zapiše razdalja prevožene poti, ki je izračunana po naslednjem algoritmu (Haversine enačba):

$$a = \sin^2\left(\frac{\Delta\varphi}{2}\right) + \cos(\varphi_1) * \cos(\varphi_2) * \sin^2\left(\frac{\Delta\lambda}{2}\right)$$

$$c = 2 * \operatorname{atan2}(\sqrt{a}, \sqrt{1-a})$$

$$d = R * c$$

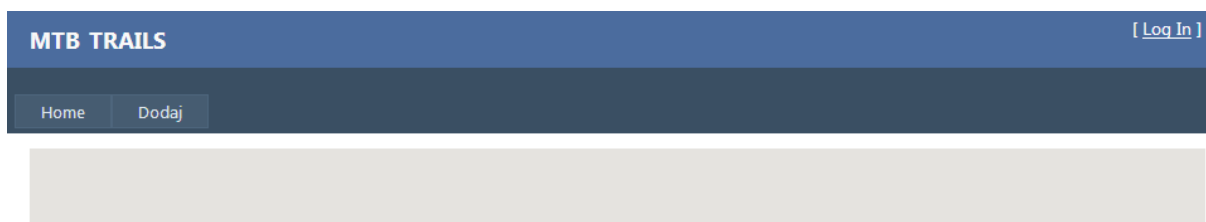
kjer je φ zemljepisna širina v radianih, λ zemljepisna dolžina v radianih, R polmer Zemlje v kilometrih ($R = 6\,371$ km) in d iskana dolžina v kilometrih.

Tako smo izračunali razdaljo med dvema točkama, ker pa je pot sestavljena iz mnogih točk, smo skupno razdaljo dobili tako, da smo posamezne razdalje sproti seštevali.

- **cas** je polje, v katerega zapišemo razliko med začetnim in končnim časom, ki je zabeležen v *.gpx datoteki.
- **zahtevnost** je polje, v katerega se zapiše izbrana vrednost iz spustnega seznama za zahtevnost.
- **opis** je polje, v katerega shranimo besedilo, ki smo ga napisali v večvrstično tekstovno okno za opis.
- **voznost** je polje, v katerega zapišemo izbrano vrednost iz spustnega seznama.
- **gpx_file_path** je polje, ki vsebuje novo nastalo ime GPX datoteke, sestavljeno iz datuma in časa.
- **added_by_user** je polje za zapis identifikacije uporabnika, ki je pot dodal.

4.3 Struktura spletne strani

Na sliki 9 vidimo, da je v ozadju naše spletne strani več spletnih form. Za spletno formo bi lahko rekli, da je to stran s HTML značkami, kontrolami in programsko kodo, ki se prevede ob uporabnikovi zahtevi v spletnem brskalniku. Do nekaterih form lahko dostopamo tako, da jih izberemo v glavnem meniju, druge se prikažejo ob kliku na hiperpovezavo (angl. Hyperlink), spet tretje pa dostopamo nezavedno, ko smo nanje preusmerjeni ob kliku na nek gumb.



Slika 9: Spletna stran z navigacijskim menijem in »Log in« kontrolo

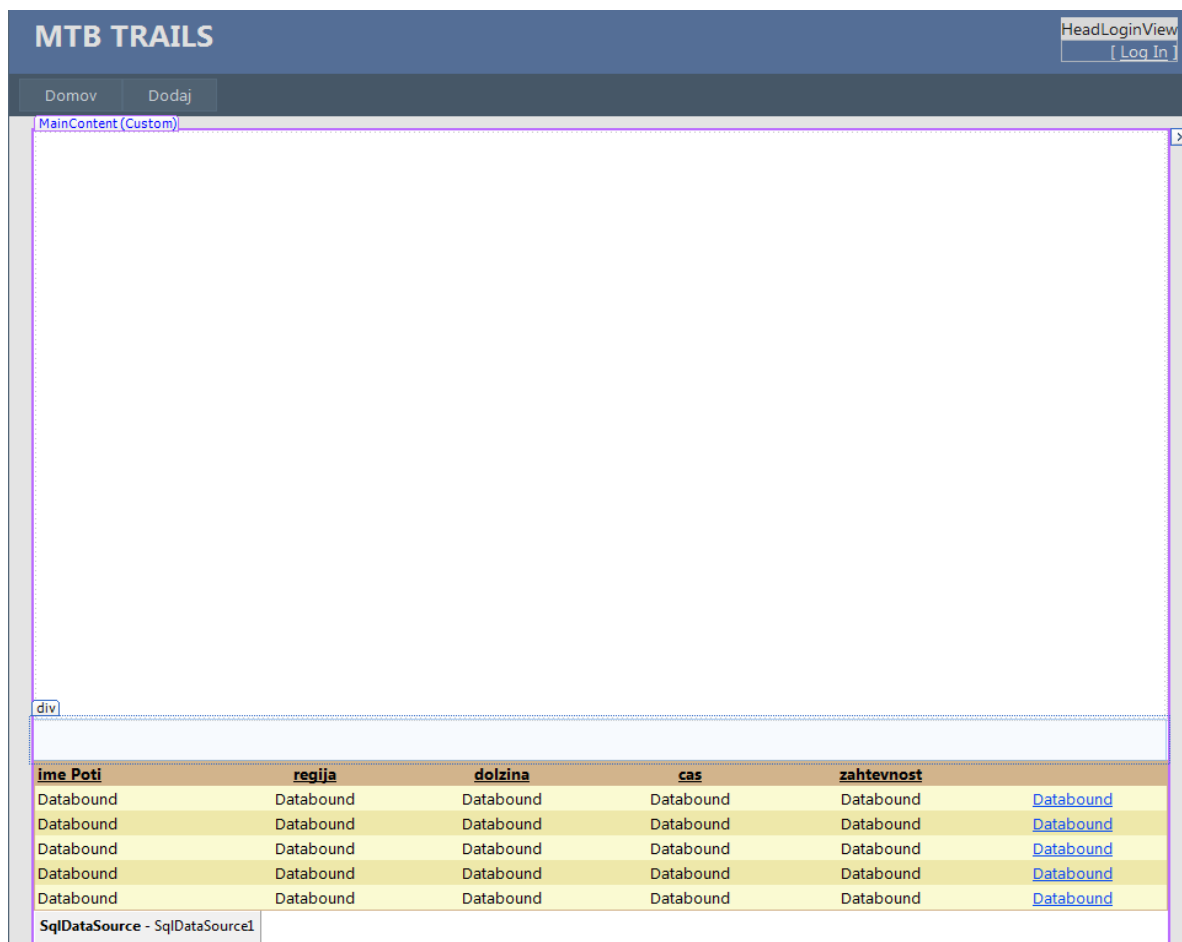
Skupna vsem spletnim formam je glavna stran (angl. Master Page), ki definira splošni izgled in strukturo strani. Na njej imamo kontrolo za prijavo (angl. Log-in control), naslov in navigacijski meni, ki ostajajo vseskozi nespremenjeni. Prijava je podobna kot jo srečujemo na veliko spletnih straneh, kjer gre za zaprte skupine ali kakšne osebne podatke. Pred prvo prijavo na spletno stran moramo vedno opraviti še predhodno registracijo, kjer si po navadi izberemo uporabniško ime in geslo ter zaupamo nekaj osebnih podatkov. »Log in« kontrola je vedno prikazana na spletni strani, ne glede na kateri formi se nahajamo, prav tako tudi naslov in navigacijski meni.

V ozadju spletne forme imamo tudi C# kodo (angl. Code behind), ki je ločena od HTML kode zaradi večje preglednosti. Torej, ločili smo kodo za prikaz spletne strani od kode za izvrševanje določenih funkcij in nalog.

V nadaljevanju bomo opisali posamezno spletno formo, kaj prikazuje, katere gradnike oz. kontrole vsebuje, kaj nam predstavljajo posamezni podatki in kaj pomenijo določene stvari in izrazi. Več informacij o posamezni kontroli lahko najdemo v Microsoft-ovi knjižnici Developer Network.

4.3.1 Spletna stran »Domov«

Kadarkoli bomo dostopali do naše spletne strani, se bo najprej prikazala domača stran »Domov«, v ozadju katere deluje spletna forma `Default.aspx` (slika 10).



Slika 10: Prikaz kontrol spletne forme Default.aspx

Zgornja forma vsebuje naslednje kontrole:

- DIV element

Na tej strani se pojavi dvakrat. Prvič smo z njim rezervirali določen prostor, v katerega smo postavili okno za interaktivno karto (velik prazen prostor na sliki 10), drugič pa ga uporabimo samo kot ločnico med karto in razpredelnico. Z njim si torej tudi pomagamo pri oblikovanju strani.

```
<div id="map-canvas" style="width: 921px; height: 500px; text-align:
left;"></div>
<div>
    <br />
</div>
```

Izpostavimo prvo `<div>` značko, kjer je definirano okno Google-ove karte z zgoraj napisanim id-jem, širino in višino okna ter poravnavo. Da bo to okno delovalo, moramo v kodo napisati tudi določene `<script>` stavke. Prvi (spodaj) naloži Google Maps JavaScript API v3.


```
<script  
src="http://maps.googleapis.com/maps/api/js?key=AIzaSyDY0kkJiTPVd2U7a  
TOAwhc9ySH6oHxOIYM&sensor=false"></script>
```

Drugi pa ustvari in konfigurira karto in je dejansko sestavljena iz dveh delov, ker funkcije `initialize()` zaradi preglednosti nismo pisali direktno na spletno formo, ampak v posebno JavaScript datoteko, na katero se sklicujemo s spodnjim stavkom:

```
<script src="Scripts/homePage.js" type="text/javascript"></script>
```

Za sprožitev same funkcije ob nalaganju strani pa smo napisali še spodnjo kodo:

```
<script type="text/javascript">window.onload = initialize</script>
```

- GridView kontrola

Je preglednica naših gorsko kolesarskih poti (slika 11), ki se avtomatsko generira na podlagi vnesenih podatkov. Podatki, prikazani v preglednici za posamezno pot, so:

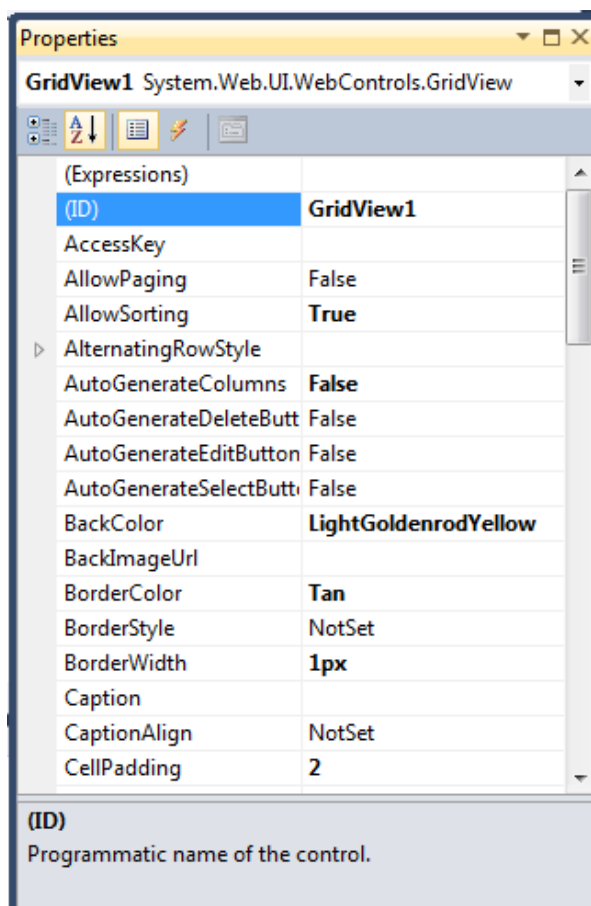
- ime poti,
- regija,
- dolžina,
- čas,
- zahtevnost,
- hiperpovezava.

ime Poti	regija	dolzina	cas	zahtevnost	
Databound	Databound	Databound	Databound	Databound	Databound
Databound	Databound	Databound	Databound	Databound	Databound
Databound	Databound	Databound	Databound	Databound	Databound
Databound	Databound	Databound	Databound	Databound	Databound
Databound	Databound	Databound	Databound	Databound	Databound

Slika 11: GridView kontrola z naslovnimi stolpci

S klikom na enega od teh naslovnih stolpcev preglednice se izvede naraščajoče ali padajoče sortiranje po izbranem atributu preglednice, razen pri hiperpovezavi, kjer to ni mogoče. Vsi podatki, ki jih vsebuje preglednica, so pridobljeni iz podatkovne baze »poti«. Pri tem moramo ustvariti povezavo med našo bazo in preglednico, kar storimo s kontrolo `SqlDataSource`, ki jo bomo opisali v nadaljevanju.

Na sliki 12 vidimo, da imamo pri tej kontroli veliko nastavitvev raznih lastnosti (angl. Properties) za prikaz, izgled in vsebino preglednice. Če kliknemo na zlato strelo, ki je prav tako razvidna iz slike, preklopimo na možnost upravljanja z dogodki (angl. Events), povezanimi s preglednico ali podatki.

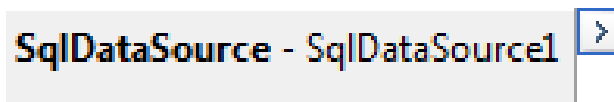


Slika 12: Možne nastavitve pri GridView kontroli

Zgoraj smo omenili hiperpovezavo, vendar nismo povedali, s čim nas poveže. Na podlagi unikatnega id-ja (id), ki ga ima vsaka pot (sled), in pa pripadajočega naslova *.gpx datoteke (gpx_file_path), nas povezava preusmeri na spletno stran oz. spletno formo opisa poti (Opis.aspx).

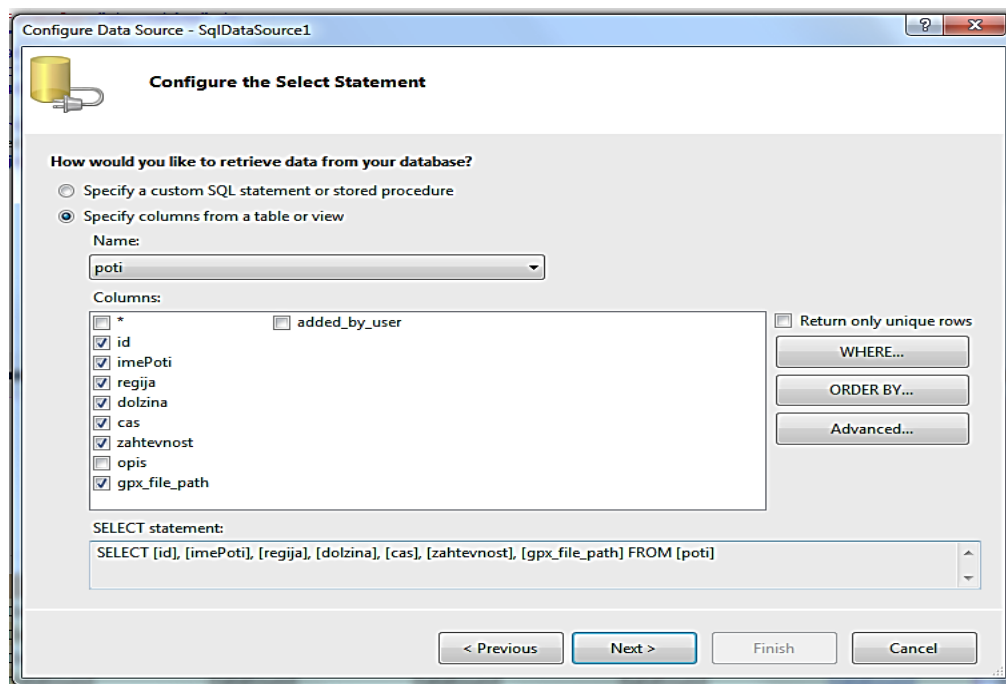
- SqlDataSource kontrola

Ta kontrola nima vizualne predstavitve na spletni strani, saj deluje v ozadju in služi za povezavo podatkovne baze z našo preglednico (GridView). V formi pa zgleda kot na sliki 13.



Slika 13: SqlDataSource kontrola za povezavo s podatkovno bazo

Iz podatkovne baze »poti« smo uporabili sledeče podatke: id, imePoti, regija, dolzina, cas, zahtevnost in gpx_file_path (slika 14).

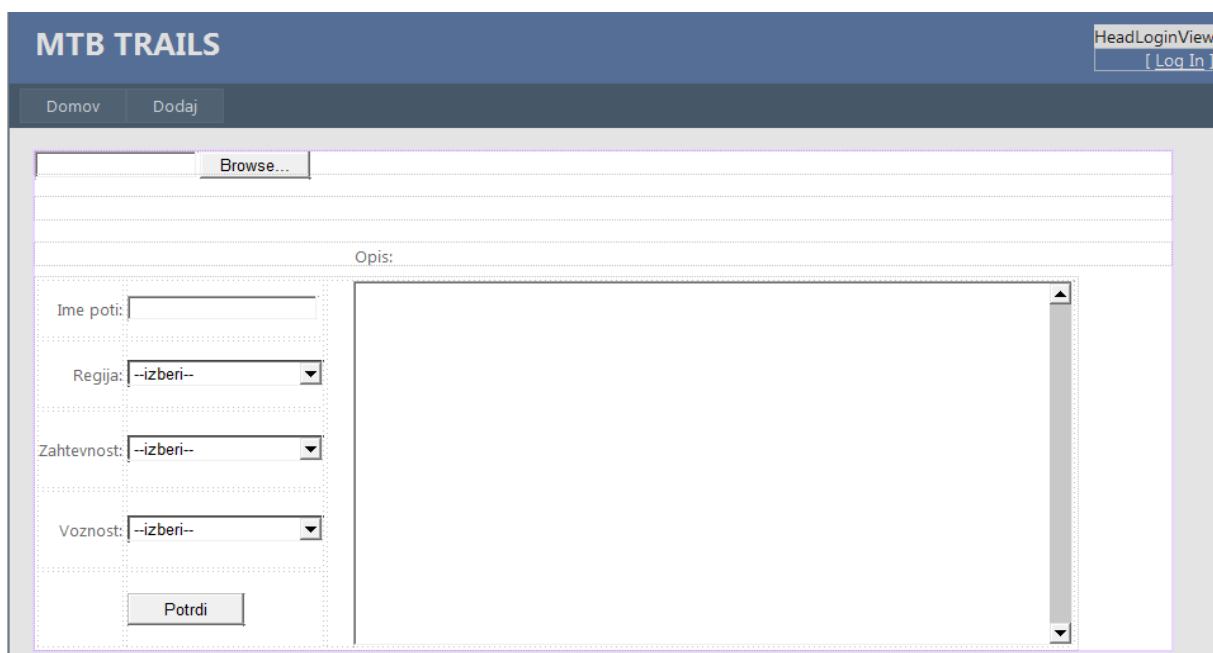


Slika 14: Konfiguriranje podatkovne povezave

Tudi pri tej kontroli imamo veliko možnosti nastavitve lastnosti in dogodkov na enak način kot pri kontroli GridView.

4.3.2 Spletna stran »Dodaj«

V ozadju te spletne strani deluje spletna forma »About.aspx« (slika 15), ki služi za dodajanje novih poti s pripadajočimi opisnimi podatki.



Slika 15: Spletna forma About.aspx

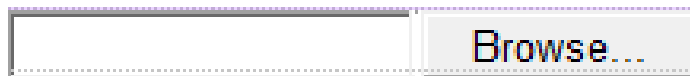
Podatki, ki jih ta spletna forma zahteva od nas, so:

- *.gpx datoteka (sled),
- ime novo nastale poti,
- regija, v kateri se pot nahaja,
- zahtevnost poti,
- voznost poti,
- opis poti.

Za vnos vseh naštetih podatkov smo uporabili naslednje kontrole:

- FileUpload kontrola

S to kontrolo je možno enostavno izbrati in naložiti *.gpx datoteko kar iz naše spletne strani. Datoteka se shrani na strežnik v mapo »Sledi«, kjer dobi tudi svoje novo unikatno ime, sestavljeno iz datuma in časa, da slučajno ne pride do podvajanja imen. Na sliki 16 vidimo izgled FileUpload kontrole. Sestavljena je iz polja za tekst, v katerem se prikaže ime izbrane datoteke, in gumba za brskanje, ki omogoča, da na svojem računalniku poiščemo želeno datoteko.



Slika 16: FileUpload kontrola

Da bo kontrola delovala, moramo v kodi, ki deluje v ozadju, napisati sledeče stavke:

```
if (FileUpload1.HasFile)
{
    FileUpload1.SaveAs(directoryPath + newPathFileName);
}
```

Zgornja koda preveri, če se je datoteka naložila, in jo v primeru uspeha shrani v določeno mapo in ji dodeli ime.

Možno je nastavljanje nekaterih lastnosti in dogodkov kontrole v oknu »Properties«. Podobno kot pri prejšnjih kontrolah in tudi pri naslednjih se ta možnost ponavlja. Razlike so samo v tem, da ene ponujajo več nastavitev in dogodkov, druge pa manj.

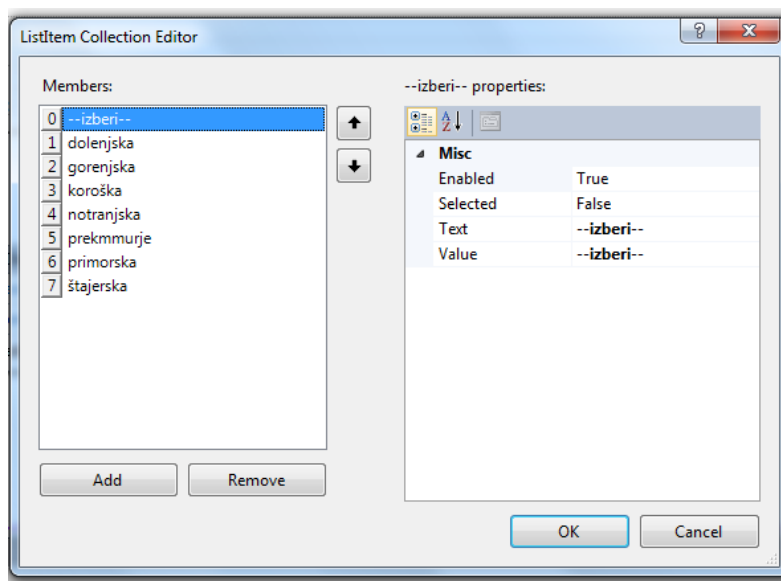
- TextBox kontrola

Omogoča, da lahko na našo spletno stran tudi kaj napišemo, ne samo beremo in brskamo po njej. V tej spletni formi je uporabljena dvakrat, prvič za vnos imena poti in drugič za opis poti. Razlikujeta se po tem, da ena omogoča vnos enovrstičnega (angl. SingleLine) teksta, druga pa

večvrstičnega (angl. MultiLine). To smo nastavili z zgoraj omenjenimi posameznimi nastavitvami lastnosti kontrol. Vizualno jo prepoznamo kot uokvirjen prazen prostor.

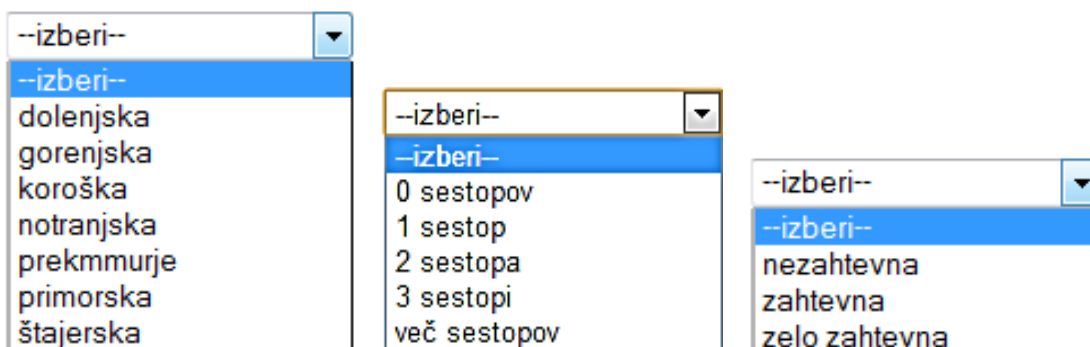
- DropDownList kontrola

Spustni seznam omogoča, da nekemu predmetu predhodno določimo člane, ki se bojo pojavili na spustnem seznamu (angl. Members), med katerimi bo uporabnik le še izbiral. To naredimo s pojavnim oknom za ustvarjanje spustnega seznama (slika 17), v katerem določimo člane in njihove lastnosti.



Slika 17: Pojavno okno »ListItem Collection Editor«

Tako smo ustvarili tri spustne sezname, enega za določitev regije, kjer se pot nahaja, drugega za težavnost oz. zahtevnost prevožene poti, in tretjega za voznost poti. Na sliki 18 vidimo ponujene člane spustnega seznama za vse tri lastnosti.



Slika 18: Spustni seznam za regijo (levo), za voznost (sredina) in za zahtevnost poti (desno)

- Button kontrola

To kontrolo uporabimo za prikaz gumba (slika 19), ki s pritiskom na le tega izvrši neko dejanje. Na splošno je to zelo uporabljana kontrola.



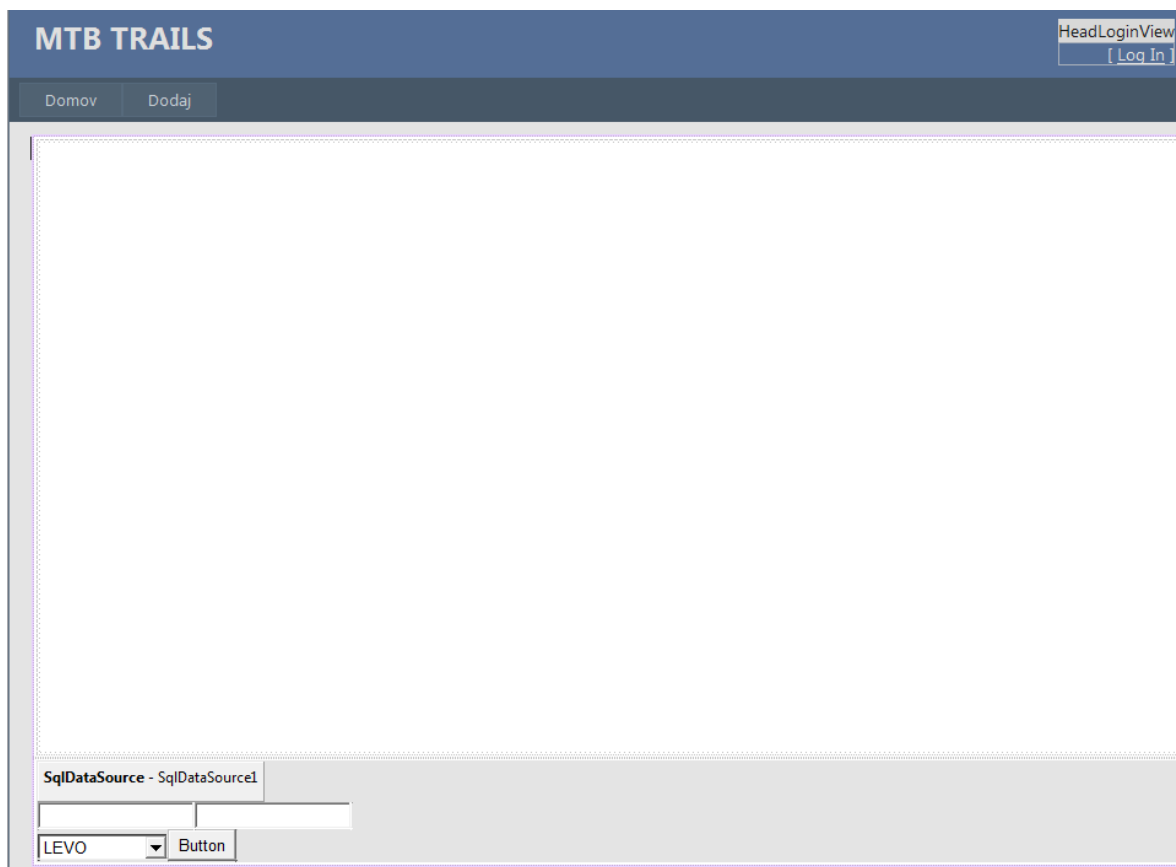
Slika 19: Kontrola Button

V tej formi smo gumb uporabili za potrditev in vnos podatkov v bazo. Podatki, ki se ob kliku na gumb zapišejo v bazo, so poleg imena, regije, zahtevnosti in opisa poti tudi dolžina, čas, naslov datoteke *.gpx in identifikacija uporabnika, ki je pot dodal. Torej v ozadju se izvrši funkcija, ki definira naslov naše datoteke, ID uporabnika, shrani datoteko na strežnik, izračuna dolžino in čas poti ter vse skupaj zapiše v bazo. Na koncu pa nas preusmeri na spletno stran za dodatni opis.

4.3.3 Spletna stran »Opis_dodatno«

Stran je namenjena dodatnemu opisu poti, konkretnije je tukaj mišljeno ustvarjanje interaktivnega kažipota. To izvedemo tako, da kliknemo na oznako, ki je postavljena na koordinate vmesne točke (angl. waypoint) in iz spustnega seznama izberemo smer (levo, desno ali naravnost) ter pritisnemo gumb »Spremeni«. Ob pritisku na gumb se v datoteko *.gpx zapiše dodaten atribut, to je spletni naslov nove ikone za smer. Hkrati se nam že prikaže nova ikona namesto oznake, tako da vemo, katero točko smo že obdelali.

Gradnike oz. kontrole, ki se pojavijo na tej strani, smo že spoznali. To so »div« element s karto, kontrola za povezavo z bazo, dva tekstovna polja, spustni seznam in gumb. V tekstovna polja se ob kliku na oznako zapišeta zemljepisna širina in dolžina, na podlagi katerih poiščemo izbrano točko v *.gpx datoteki, ki ji dodamo atribut »url«. Pridobimo tudi na preglednosti in kontroli izbrane oznake.



Slika 20: Spletna forma Opis_dodatno.aspx

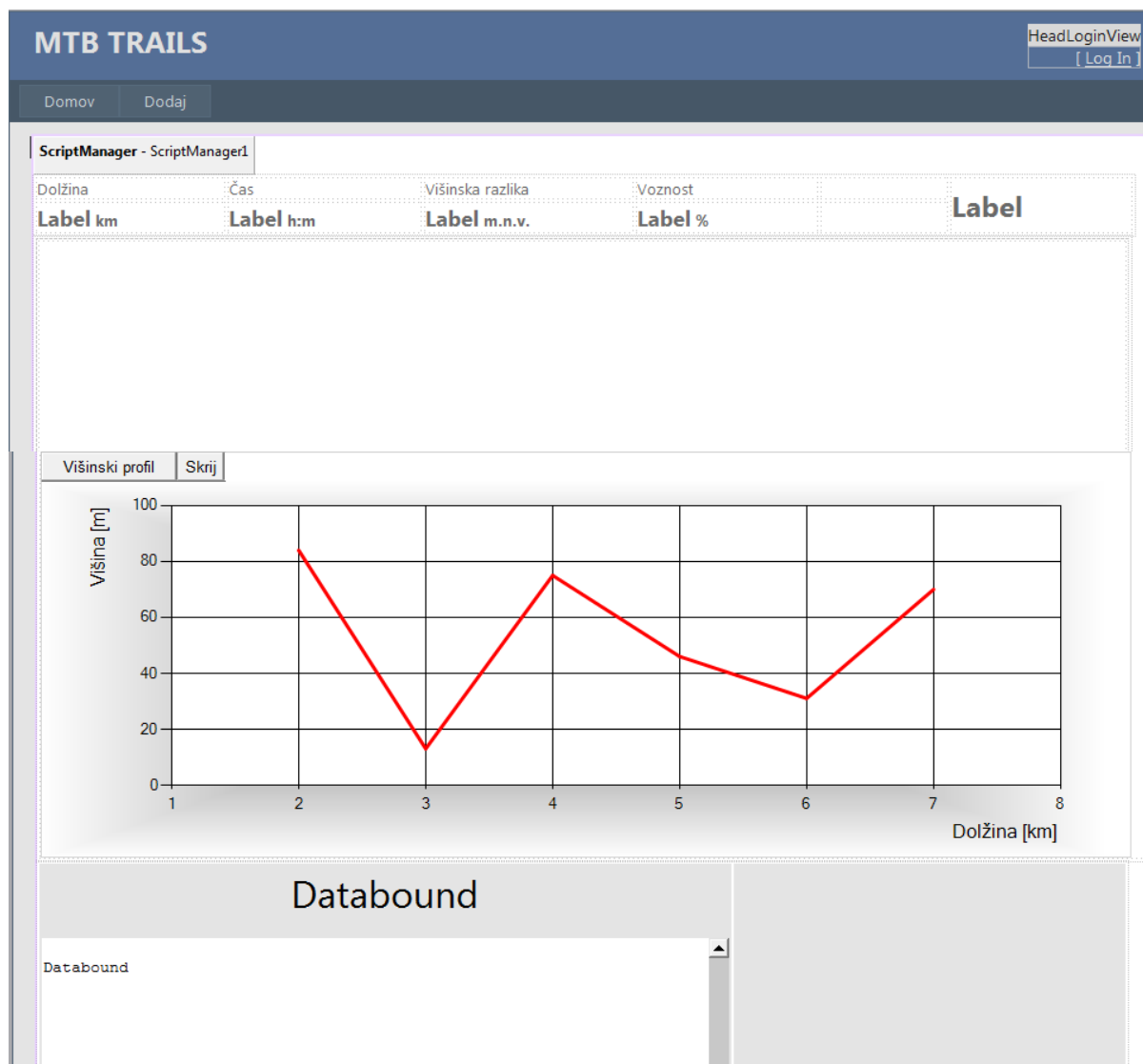
Na sliki 20 vidimo ozadje spletne strani za dodatni opis, to je spletna forma Opis_dodatno.aspx z zgoraj naštetimi kontrolami.

4.3.4 Spletna stran »Opis«

»Opis« je najbolj obsežna stran, saj vsebuje vse podatke, ki nas zanimajo o neki poti. Do nje dostopamo preko hiperpovezave v preglednici ali pa preko oznake na karti na domači strani.

V ozadju te spletne strani deluje spletna forma Opis.aspx (slika 21), ki vsebuje kar nekaj kontrol. Nekaj jih že poznamo, nekaj jih bomo še spoznali v nadaljevanju. Poznamo že okno karte, kontrolo za gumb in kontrolo za povezavo s podatkovno bazo. Na novo pa se pojavijo naslednje kontrole:

- Label (označba),
- Chart (graf),
- FormView (formular),
- ScriptManager (upravljalca skript),
- UpdatePanel (posodobitvena plošča).



Slika 21: Spletna forma Opis.aspx

- Label kontrola

Kontrola omogoča prikaz teksta na strani, ki ni nujno ročno napisan v polje kontrole, ampak se lahko programsko zapiše od nekod. Tako imamo na naši spletni formi pet omenjenih kontrol, in sicer za prikaz dolžine poti, časa, višinske razlike, voznosti in imena poti. Vse te kontrole, razen višinske razlike, dobijo podatek iz podatkovne baze, ki se avtomatsko zapiše v polje kontrole. Višinsko razliko pa najprej izračunamo v kodi, ki je napisana v ozadju te spletne forme, in jo nato zapišemo v označbo. Primer zapisa teksta v neko označbo v programski kodi poteka, kot je napisano v spodnjem stavku:

```
Label1.Text = myReader["dolzina"].ToString();
```

Sam izgled kontrole vidimo na sliki 21, kjer piše »Label«.

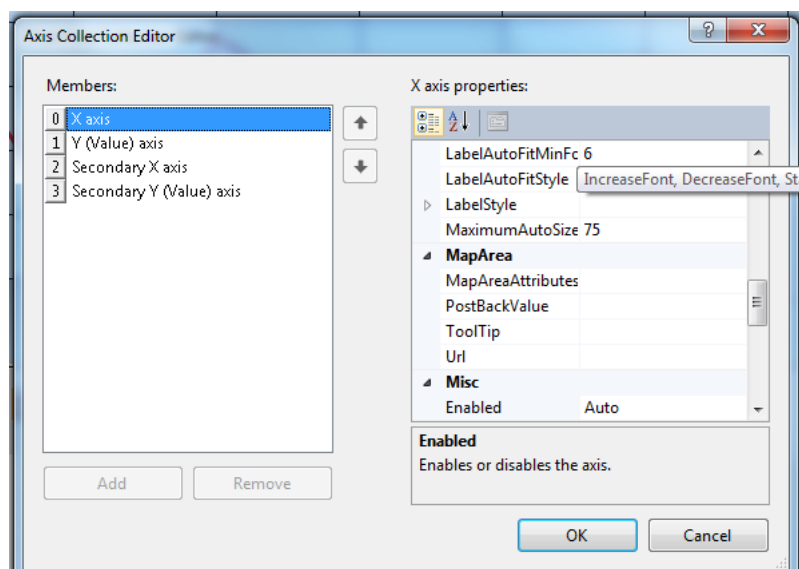
- Chart kontrola

Ta gradnik (slika 22) omogoča postavitev grafa na spletno stran. Uporabimo ga za grafični prikaz številskih podatkov, in sicer za prikaz višine v odvisnosti od dolžine.



Slika 22: Graf višinske razlike

Iz slike lahko razberemo, da gre za linijski graf, kjer os X predstavlja dolžino poti v kilometrih, os Y pa višino točke (v metrih), kjer se nahajamo v trenutku sprejema GPS signala. Interval dolžine in višine se samodejno spreminja za vsako pot posebej, odvisno od njene dolžine/višine.



Slika 23: Pogovorno okno »Axis Collection Editor«

Ta kontrola nam ponuja mnogo nastavitev za prilagajanje grafa našim potrebam. Tiste, ki smo jo uporabili, vidimo na sliki 23, kjer imamo pogovorno okno za urejanje primarnih osi grafa. Tukaj smo poimenovali osi in določili intervale le teh.

- FormView kontrola

Uporablja se za prikaz enkratnega zapisa, pridobljenega iz izbrane podatkovne zbirke. Z njo ustvarimo predlogo za prikaz in urejanje podatkovnih vrednosti. Ta predloga dejansko vsebuje več kontrol, povezanih med seboj, ki v končni obliki predstavljajo neko celoto - formo.



Slika 24: FormView kontrola v načinu urejanja predloge

Slika 24 prikazuje grobo sestavo naše kontrole. Razberemo lahko dva že znana gradnika te kontrole, to sta označba (»Label«) za prikaz imena poti, katero opisujemo, in več vrstično tekstovno polje (»TextBox«) za prikaz samega opisa. Ime in opis poti pridobimo iz podatkovne baze »poti«. S tem smo torej omogočili prikaz podrobnejšega opisa poti.

- Kontrolni ScriptManager in UpdatePanel

Sta Microsoft-ovi AJAX strežniški kontrolni, ki uporabniku pričarata lepše doživetje ob brskanju po konkretnih vsebinah na trenutni spletni strani. Drugače povedano, ni nepotrebnega osveževanja celotne strani pri klikanju po raznih vsebinah na strani.

Slika 25: ScriptManager kontrola

Praktično ti dve kontroli uporabimo tako, da znotraj njiju postavimo tiste kontrole, za katere želimo, da se samostojno osvežijo brez ponovnega nalaganja strani. Kako se to napiše v programski kodi, lahko vidimo spodaj:

```
<asp:ScriptManager ID="ScriptManager1" runat="server">
</asp:ScriptManager>
<asp:UpdatePanel ID="UpdatePanel1" runat="server">
  <ContentTemplate>
    <asp:Button ID="Button1" runat="server" onclick="Button1_Click"
      Text="Višinski profil" />
    <asp:Button ID="Button2" runat="server" onclick="Button2_Click"
      Text="Skrij" />
    <asp:Chart ID="Chart1" runat="server" BackColor="DimGray"
    </asp:Chart>
  </ContentTemplate>
</asp:UpdatePanel>
```

V kodi so napisane tri kontrole (dva gumba in graf), ki se izvajajo znotraj kontrole »UpdatePanel«, torej brez osveževanja celotne strani.

5 NAVODILA IN POJASNILA UPORABNIKOM

V tem poglavju je predstavljeno delovanje spletne strani, v večji meri namenjeno tistim, ki nameravajo dodajati nove poti. Zato bomo navodila razdelili na preproste in napredne uporabnike. Ti so tako že ločeni pri zasnovi strani, saj preprosti uporabniki za potrebe brskanja po strani oz. ogledovanja že naloženih poti ne potrebujejo registracije in prijave na spletno stran, medtem ko se morajo napredni uporabniki registrirati ob prvi prijavi in prijaviti pri vsakem vnašanju novih poti.

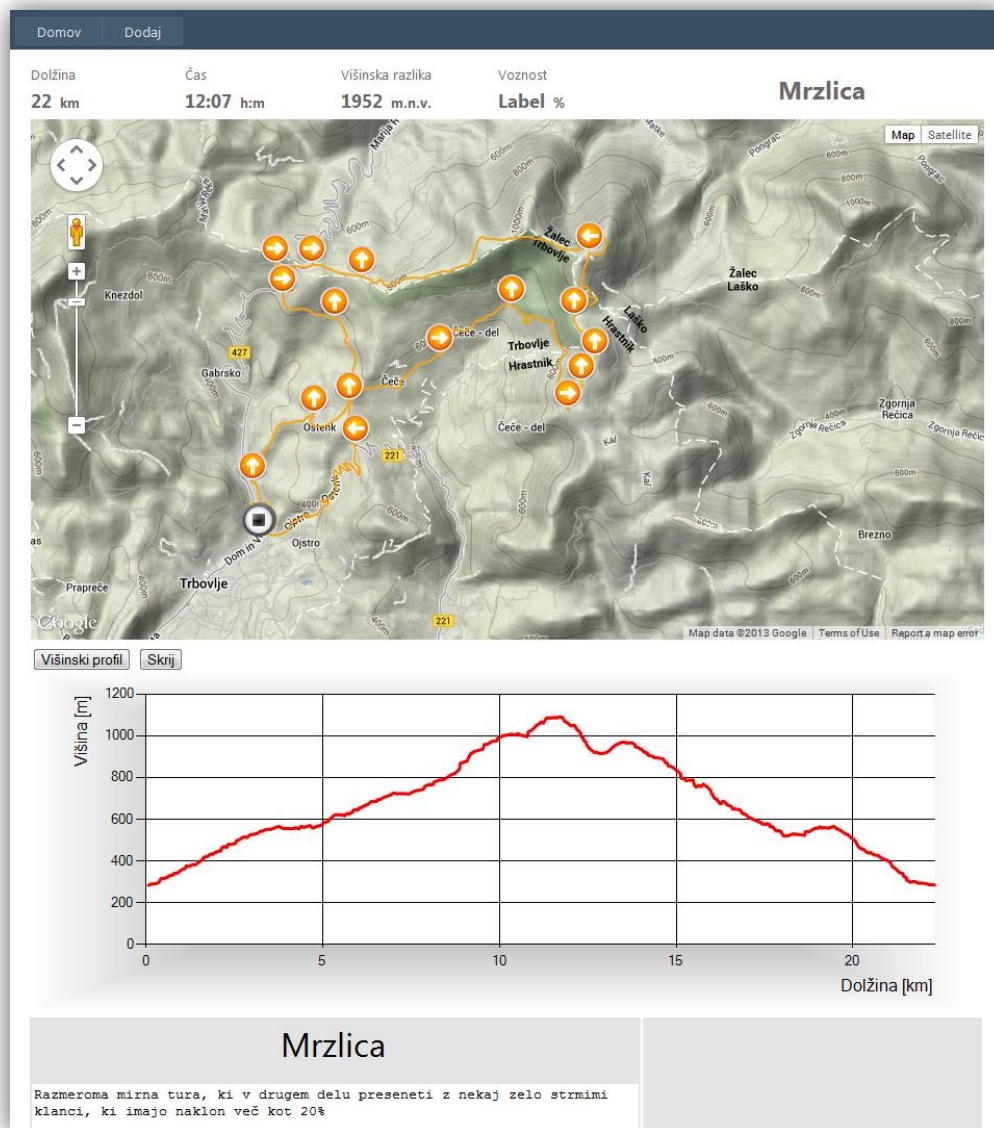
5.1 Navodila za preproste uporabnike

Pri dostopanju na spletno stran se odpre domača stran »Domov«, kjer so na karti označene poti z oznakami, in pa preglednica vseh poti v bazi. Oznaka na karti predstavlja izhodiščno točko poti. Karta oz. uporabniški vmesnik Google maps API je lahko prikazan v slovenskem ali angleškem jeziku, odvisno od nastavljenega jezika v vašem spletnem brskalniku. Pot lahko izberemo na karti ali pa v preglednici, bodisi s klikom na oznako bodisi s klikom na hiperpovezavo »select«. Za izbiro poti po določenih kriterijih je v preglednici možno sortirati vnose po izbranem atributu.

ime Poti	regija	dolzina	cas	zahtevnost
Brnica	štajerska	14	01:26:00	zahtevna select
Pod Stolom	gorenjska	22	00:06:00	zelo zahtevna select
Mrzlica	štajerska	22	00:07:00	zahtevna select

Slika 26: Domača stran naše spletne strani

Na sliki 26 vidimo domačo stran, iz katere smo ob kliku na izbrano pot preusmerjeni na stran z opisom poti, ki jo vidimo na sliki 27.



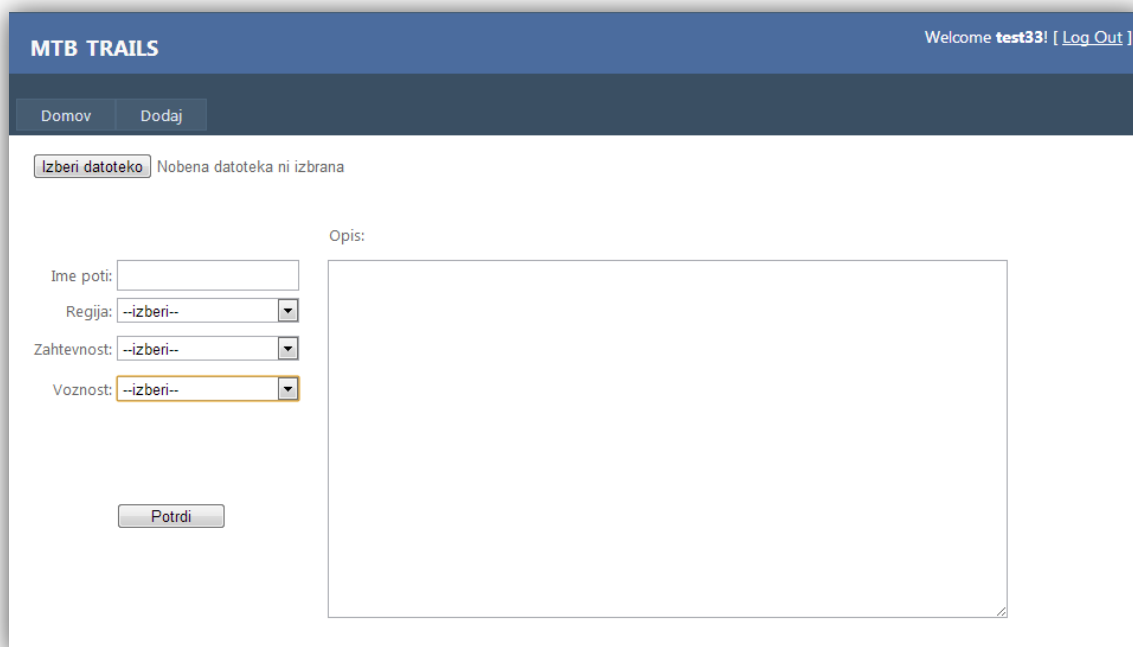
Slika 27: Stran z opisom izbrane poti

Na tej strani imamo na vrhu napisane podatke o dolžini, času, višinski razliki (seštevek vseh vzponov) in voznosti poti ter ime poti. Potem je prikazana interaktivna karta s kaŕipotom, kjer puščice pomenijo ali je treba zaviti levo, desno ali naravnost. Tukaj bi opozorili na smer puščice, ki včasih kaže v smeri narisane sledi, ampak naj vas to ne zavede. Če je puščica, npr. obrnjena navzgor, to ne pomeni, da morate peljati v tisti smeri, ampak pomeni, da je v kriŕiŕu potrebno peljati naravnost. Lahko pride tudi do prekrivanja začetne in končne točke, če imamo kroŕno turo. To prepoznamo tako, da vidimo narisano samo oznako za konec poti. Problem se običajno odpravi s povečavo pogleda, kjer se nato prikaŕeta začetna in končna označba (znak za »play« in »stop«) poti. V tem primeru je potrebno

podlago karte preklopiti na satelitske posnetke ali na avtokarto, ki omogočata večjo stopnjo povečave. Pod karto imamo možnost prikaza višinskega profila poti, ki ga pogledamo s klikom na gumb »Višinski profil« in skrijemo s klikom na gumb »Skrij«. Prikaže se še podrobnejši opis poti.

5.2 Navodila za napredne uporabnike

V to skupino spadajo tisti, ki bodo dodajali nove gorsko kolesarske poti na spletno stran. V navigacijskem meniju pritisnete gumb dodaj in preusmerjeni boste na stran za vnos podatkov. V primeru, da še niste prijavljeni, se vam prikaže »Log in« kontrola, kjer se prijavite. Podatki za vnos nove poti, ki se od vas zahtevajo, so ime, regija, zahtevnost, voznost in podrobnejši opis poti. Izbrati morate še *.gpx datoteko, ki jo želite naložiti. Vse to je prikazano na sliki 28.



The screenshot shows the 'MTB TRAILS' website interface. At the top, there is a navigation bar with 'Domov' and 'Dodaj' buttons. Below this, a message says 'Izberi datoteko' and 'Nobena datoteka ni izbrana'. The main form contains the following elements:

- Ime poti:
- Regija:
- Zahtevnost:
- Voznost:
- Opis:
- Potrdi:

Slika 28: Stran za dodajanje nove poti

Pri dodajanju *.gpx datoteke moramo biti pozorni na vsebino datoteke, saj mora le ta vsebovati v znački <gpx> naslednji znakovni niz:

```
xmlns="http://www.topografix.com/GPX/1/0"
```

To preverite tako, da datoteko odprete z beležnico in pogledate, če se ta zapis kje nahaja, drugače ga ročno napišite in shranite datoteko. Bodite pozorni, da na koncu piše /1/0 in ne /1/1, ker gre za različne verzije GPX-a. Če bo vaša datoteka že vsebovala zgornji niz, je odvisno od proizvajalca GPS sprejemnika in njegove programske opreme.

Zahtevnost ocenite na podlagi naklona, dolžine in višinske razlike. Spodaj so napisane okvirne ocene:

- *nezahtevna*: naklon do 10 %, dolžina do 10 km, višinska razlika do 400 m
- *zahtevna*: naklon do 20 %, dolžina do 20 km, višinska razlika do 800 m
- *zelo zahtevna*: naklon nad 20 %, dolžina nad 20 km, višinska razlika nad 800 m

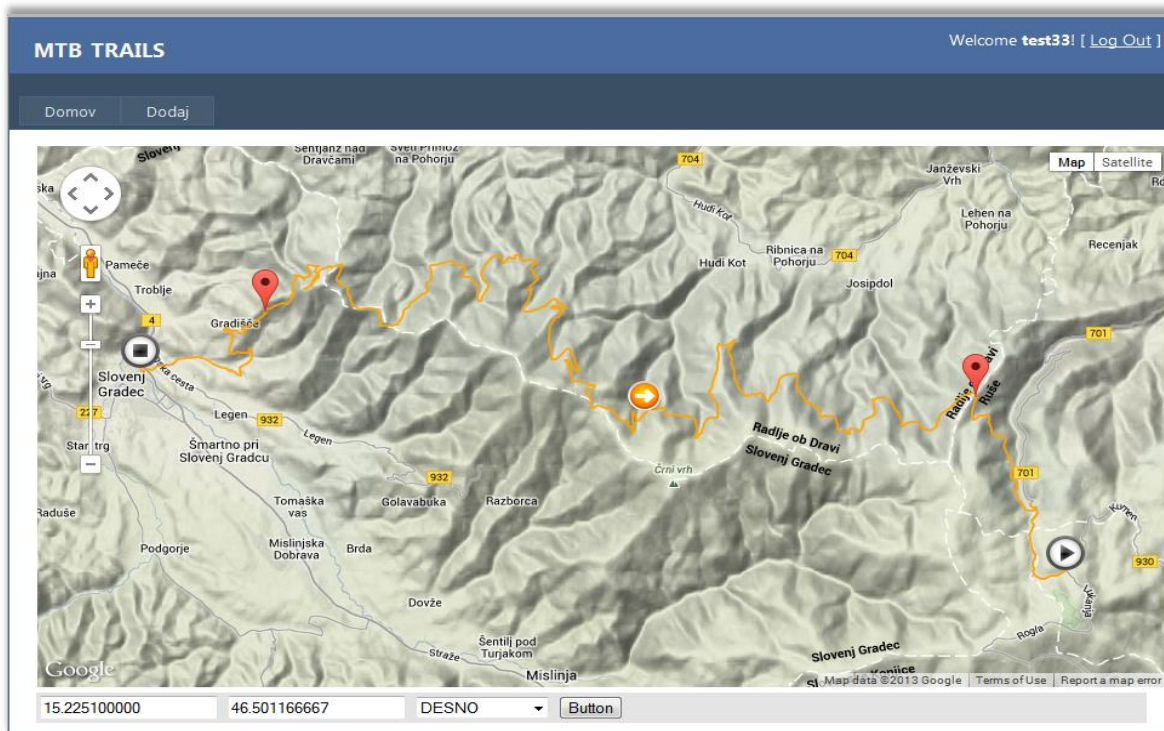
Po teh ocenah se je dejansko težko dosledno orientirati, saj se dejavniki med seboj prepletajo in odvisno je tudi v kolikšni meri se pojavljajo. Lahko imamo zelo zahtevno pot, ki je dolga do 10 km, vendar sta naklon in višinska razlika velika. Pomembno je tudi, za kakšen teren gre, lahko je to vozna gozdna pot brez večjih korenin, lahko z veliko koreninami in skalami, z blatom itd... Zato bo tukaj potrebno tudi nekaj subjektivne ocene, glede na nek kompromis med fizično in tehnično zahtevnostjo poti. Če menite, da lahko to pot prevozi skoraj vsak kolesar, potem naj bo to nezahtevna pot, če se ta množica zmanjša, jo ocenite kot zahtevno, če pa jo lahko premagajo le »pravi« gorski kolesarji, naj bo to zelo zahtevna pot. Seveda imejte pri tem v mislih zgornje dejavnike.

Voznost poti izberete na podlagi števila sestopov s kolesa. Izbirate lahko med vnosi 1 sestop, 2 sestopa, 3 sestopi ali več sestopov. Sestope lahko podrobneje opišete v polju za opis, npr. za kakšno oviro gre, ali je daljši ali krajši sestop itd. Za sestop s kolesa se šteje, ko je ovira ali pot res nepremostljiva oz. neprevozna. Torej če menite, da lahko tehnično in fizično boljši kolesar prevozi to pot ali oviro brez sestopa, potem je pot 100 % vozna.

Pri opisu poti opišite razne značilnosti ture, kaj se pojavlja ob poti, omembe vredni vzponi in spusti, razne prepreke, zanimivosti itd.

Ko smo zaključili z vnosom podatkov, pritisnemo na gumb potrdi in odpre se nam nova stran za ustvarjanje interaktivnega kažipota. Zato je potrebno že pri snemanju poti zabeležiti točke križišč, saj se nam bodo le v tem primeru prikazale oznake na karti. Sistem je tak, da pritisnemo na oznako, ob tem pa se nam v tekstovnih poljih izpišejo koordinate oznake. Nato izberemo iz spustnega seznama pravo smer ter pritisnemo gumb »Spremeni«. Stran se osveži in že se ikona oznake spremeni v izbrano puščico, kot je na sliki 29. Postopek ponavljamo, dokler ne spremenimo vseh oznak v ikone, ki predstavljajo smer. Ko končamo, se vrnemo na stran »Domov«.

Na strani za ustvarjanja kažipota lahko vidimo tudi, če je prišlo do morebitnih napak pri sprejemu GPS signala v času snemanja poti, npr. izgube signala ali osamele točke. V tem primeru obvestimo upravljavca spletne strani, ki ima dostop do naložene datoteke, le ta pa poišče in popravi napačne točke. Če gre za daljši izpad signala, pa je smiselno še enkrat posneti celotno pot.



Slika 29: Stran za ustvarjanje kažipota

6 ZAKLJUČEK

V diplomski nalogi smo predstavili in izdelali eno od možnih rešitev za problem, to je razrešili dvome kolesarja, kam se podati na turo, kam zaviti, koliko časa potrebuje za pot, itd. Izdelana spletna karta nam tako omogoča, da se seznanimo s terenom, po katerem nameravamo kolesariti. Bistvene značilnosti našega izdelka, h katerim smo stremeli, so enostavnost, preglednost in hitrost iskanja ter prebiranja informacij. Poskušali smo doseči, da se bo uporabnik čim bolj znašel na terenu iz pridobljenih informacij na internetu; predvsem to, kje in kam je treba zaviti, da ostane na pravi poti, v ta namen je narejen interaktivni kašipot, ali bo zmožen prekolesariti pot, kar se navezuje na podatke o dolžini in zahtevnosti poti ter višinski razliki, pa tudi o voznosti poti, ter pomemben je čas, saj je dan danes »čas denar«.

Naša spletna stran ne omogoča samo prikaza poti na karti in branja opisnih podatkov, ampak tudi dodajanje novih kolesarskih poti. To smo izvedli na način, ki omogoča enostaven vnos podatkov s čim manj pisanja le teh in s čim več pridobljenimi podatki iz GPX datoteke, ki smo jo posneli z GPS sprejemnikom na terenu. Na ta način smo mogoče spodbudili nekatere kolesarje v odkrivanje novih poti, vsekakor pa omogočili obsežnejšo bazo gorskih kolesarskih poti. S tem se del uporabnosti in odgovornosti o točnosti podatkov prenese na same uporabnike, saj bodo le ti dodajali nove podatke, na kar se od njih pričakuje, da bodo podatki tudi verodostojni. Ker gre za vnos podatkov s strani različnih uporabnikov, bi tukaj omenil še subjektivnost uporabnika. Recimo že pri določanju zahtevnosti poti se pojavi problem, kam turo uvrstiti, saj nismo podrobneje obdelali te postavke, ker je tukaj veliko dejavnikov, ki bi jih morali upoštevati in kombinirati za primerno oceno zahtevnosti poti.

Za dodajanje novih poti na spletno stran sta se v začetku ponudili dve možni izvedbi tega problema. Prvi način je ta, da podatke vedno preberemo neposredno iz *.gpx datoteke, kar se je v začetku zdelo enostavno vendar se je izkazalo za ne najbolj elegantno rešitev, vsaj iz vidika programerjev. Drugi pa je ta, da vse točke in njihove koordinate shranimo v podatkovno bazo in jih preberemo ob nalaganju karte. Ker smo izbrali prvi način, smo delno morali uporabljati tudi drugega, s tem, da nismo podatkov shranili v bazo ampak jih samo uporabili za izračune posameznih vrednosti, kot sta npr. dolžina in čas poti.

Spletna stran zaenkrat deluje lokalno in je narejena za namen izdelave diplomske naloge in ne za širšo uporabo, saj bi bilo potrebno dodelati nekatere stvari glede združljivosti, varnosti in pa same vsebine karte in opisnih podatkov. Ena možnost nadgradnje, ki bi oplemenitila našo spletno stran, je dodajanje slik, ki bi se navezovala na določeno oznako na karti, recimo sliko križišča kjer bi morali zaviti ali pa zgolj na slike znamenitosti, ki se pojavljajo ob poti. Potem bi lahko dodali interaktivni prikaz zahtevnosti odseka poti in kategorizacijo odsekov, ali gre za pešpot, kolovoz, makadam ali asfalt, dodali bi lahko tudi aktualno stanje poti, če je slučajno kje podrt drevo na poti ali kakšen sprožen plaz, ki onemogoča prehodnost poti itd. S kategorizacijo bi se razširila tudi sama uporabnost aplikacije

na širšo skupino kolesarjev in ne samo gorskih kolesarjev. Zelo dobrodošla bi bila tudi možnost prenosa *.gpx datoteke na lokalni računalnik in nalozitev na lastni GPS sprejemnik.

VIRI

Bonačić, D. 2008. Okenske aplikacije z grafičnim vmesnikom v okolju MS Visual Studio: orodja za razvoj aplikacij: zbrano gradivo za predavanja. Maribor, Fakulteta za elektrotehniko, računalništvo in informatiko, Inštitut za informatiko: 39 str.

Glenn, J. 2003. ASP.NET in 60 minutes a day. Indianapolis, Wiley: 772 str.

Kaltenekar, M. 2006. Oblikovanje spletnih strani: HTML, CSS in JavaScript: hitri vodnik. Ljubljana, Pasadena: 217 str.

Peterca, M., Radovan, D., Rojc, B. 2001. Matematična kartografija: kartografske projekcije. Ljubljana, Fakulteta za gradbeništvo in geodezijo: Geodetski inštitut Slovenije: 211 str.

Ajax. 2013.

[http://sl.wikipedia.org/wiki/Ajax_\(programiranje\)](http://sl.wikipedia.org/wiki/Ajax_(programiranje)) (Pridobljeno 14. 7. 2013.)

API. 2013.

http://en.wikipedia.org/wiki/Application_programming_interface (Pridobljeno 14. 7. 2013.)

ASP.NET. 2013.

<http://www.asp.net/get-started> (Pridobljeno 8. 10. 2013.)

Google Developers. 2103. Google Maps API.

<https://developers.google.com/maps/documentation/javascript/tutorial> (Pridobljeno 5. 3. 2013.)

History. 2013.

http://en.wikipedia.org/wiki/Google_Maps (Pridobljeno 5. 3. 2013.)

Mercator projection. 2013.

http://en.wikipedia.org/wiki/Mercator_projection (Pridobljeno 15. 7. 2013.)

Microsoft Developer Network. 2013.

[http://msdn.microsoft.com/en-us/library/3xdhey7w\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/3xdhey7w(v=vs.85).aspx) (Pridobljeno 22. 8. 2013.)

Microsoft SQL Server. 2013.

http://en.wikipedia.org/wiki/Microsoft_SQL_Server (Pridobljeno 18. 7. 2013.)

Schwartz, B. 2011. Google Drops Tele Atlas & Uses Their Own Maps Data For UK Maps. SearchEnginLand.com.

<http://searchengineland.com/google-drops-tele-atlas-uses-their-own-maps-data-for-uk-maps-104107> (Pridobljeno 20. 7. 2013.)

Security concerns. 2013.

http://en.wikipedia.org/wiki/Google_Maps (Pridobljeno 5. 3. 2013.)

Topografix, 2013. GPX: The GPS Exchange Format.

<http://www.topografix.com/gpx.asp> (Pridobljeno 6. 3. 2013.)

W3Schools, 2013.

<http://www.w3schools.com/> (Pridobljeno 5. 3. 2013.)

XML. 2013.

<http://en.wikipedia.org/wiki/XML> (Pridobljeno 6. 3. 2013.)

Zaversnik, M. 2013. JavaScript. Ljubljana: Fakulteta za matematiko in fiziko.

<http://zaversnik.fmf.uni-lj.si/Gradiva/JavaScript/> (Pridobljeno 14. 7. 2013.)