

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Tarkvarateaduse instituut

Siret Sarv 142358IAPB

**AUTOMATISEERITUD
REISIPLANEERIMISE VEEBIRAKENDUS**

Bakalaureuse töö

Juhendaja: Roger Kerse

Magister

Tallinn 2016

Autorideklaratsioon

Kinnitan, et olen koostanud antud lõputöö iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

Autor: Siret Sarv

Annotatsioon

Antud lõputöö põhieesmärk on luua automatiseeritud reisiplaneerimise veebirakendus, kus on võimalik leida mitme sihtkohaga lennureise automatiseeritud kujul. Rakenduses on kasutatud programmeerimiskeelt PHP ning lennupiletite otsinguks Skypicker API-d.

Töö alguses kirjeldatakse veebirakenduse olemust ja selle turvalisust. Seejärel annab autor ülevaate olemasolevatest lahendustest ning nende võimalustest. Samuti on kirjeldatud ka antud veebirakenduse arenduskäiku ning loomisprotsessi.

Töö tulemuseks on veebirakendus, millega saab leida automatiseeritud mitme sihtkohaga lennureisi. Peale selle saab otsida ka ühe sihtkohaga lennureise, kus on võimalik otsida ka kõiki lennureiside võimalusi sihtkohta lisamata.

Antud lõputöö on leidav aadressil:
<http://dijkstra.cs.ttu.ee/~Siret.Sarv/kyklos/site/action/dashboard.php>

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 31 leheküljel, 6 peatükki, 8 joonist, 1 tabelit.

Abstract

Automated travel planning web application

The thesis „Automated travel planning web application“ main objective is to create automated travel planning web application, where you can find multiple destination airfares in automated form. This application is written in PHP programming language and uses Spypicker API for finding airfare.

In the beginning of this thesis author describes what is a web application and how can it be secured. Then the author gives an overview of available solutions. There is also description of the course of web application development and creation process. Lastly author analyzes its work and explains whether this work achieved its objectives.

The result of this thesis is a web application which you can use for finding automated multi destination air travel. In addition, you can also search for one destination airfares, where you can search for air travel if you do not know exact destination yet.

This thesis can be found at:
<http://dijkstra.cs.ttu.ee/~Siret.Sarv/kyklos/site/action/dashboard.php>

The thesis is in Estonian and contains 31 pages of text, 6 chapters, 8 figures, 1 tables.

Lühendite ja mõistete sõnastik

ATI	TTÜ Arvutitehnika instituut
DPI	<i>Dots per inch</i> , punkti tolli kohta
API	<i>Application Program Interface</i> . Rakendusliides; rakendustarkvara liides
PHP	<i>Hypertext Preprocessor</i> . Skriptimiskeel, mida kasutatakse peamiselt serveripoolsetes lahendustes dünaamiliste veebilehtede loomisel (Vikipeedia 2015).
HTML	<i>Hypertext Markup Language</i> . Hüperteksti märgistuskeel
CGI	<i>Common Gateway Interface</i> . Ühine lüüsiliides
CSS	<i>Cascading Style Sheets</i> . Kaskaadlaadistik.
HTTP	<i>Hypertext Transfer Protocol</i> . Hüperteksti Ülekande Protokoll. Standarite protokoll, mis võimaldab kasutajatel veebilehekülgedel informatsiooni vahetada.
Git	Vaba hajutatud versioonihaldustarkvara
JavaScript	Lihtsam ja piiratum kui Java programmeerimiskeel, mis võimaldab Java objekte manustada vahetult HTML lehekülgedele
jQuery	Javascripti teek
AJAX	<i>Asynchronous JavaScript and XML</i> . Asünkroonne JavaScript ja XML.
XML	<i>Extensible Markup Language</i> . Moodus kirjeldamiseks ja kinnitamaks andmeid XML keskkonnas.
Library	Teek. Failide, programmide, tavakäskluste kogum.

Sisukord

Sisukord.....	6
1 Sissejuhatus	10
2 Veebirakendused	11
2.1 Veebirakenduste eelised ja puudused	11
2.2 Veebirakenduste turvalisus.....	12
3 Olemasolevad lahendused	13
3.1 Olemasolevate lahenduste analüüs	14
4 Reisiplaneerimisrakenduse arendus.....	16
4.1 Arenduskäik.....	16
4.2 Arendamisel kasutatud tehnoloogiad.....	17
4.2.1 CGI	17
4.2.2 Skypicker API ja Google Maps API	17
4.2.3 PHP.....	18
4.2.4 JavaScript ja jQuery	19
4.2.5 Bootstrap.....	20
4.2.6 Twig.....	20
4.3 Arendamisel kasutatud tööriistad	20
4.3.1 PhpStorm	20

4.3.2 GIT	21
5 Valminud rakendus.....	22
5.1 Mitme sihtkohaga lennureisi otsingu vaade	23
5.2 Ühe sihtkohaga lennureisi otsingu vaade	25
5.3 Leitud reisi detailvaade.....	25
6 Reisiplaneerimisrakenduse analüüs	27
7 Kokkuvõte	29
Kasutatud kirjandus	30

Jooniste loetelu

Joonis 1. CGI programmi töötamise joonis	17
Joonis 2. JavaScript Google Mapsi kaardi loomiseks	18
Joonis 3. PHP Requests päring sihtkohaotsimiseks	19
Joonis 4. Google Maps kaart sihtkoha märgiga.....	22
Joonis 5. Mitme sihtkohaga otsingu blokk	23
Joonis 6. Mitme sihtkohaga otsingu tulemus	24
Joonis 7. Ühe sihtkohaga lennureisi otsingu blokk	25
Joonis 8. Hüplikakna vaade	26

Tabelite loetelu

Tabel 1. Olevasolevate lahenduste võrdlus	15
---	----

1 Sissejuhatus

Reisimine on muutunud inimeste seas üha populaarsemaks, ning aina tihedamini kasutatakse reisi planeerimisel veebi otsingusüsteemide abi. Nii hoitakse kokku aega ja raha, mis kuluks reisibürooga suhtlemisele. Enamasti on lennupiletite otsimine reisi planeerimise lahutamatu osa. Lennupiletite otsingusüsteemide on olemas mitmeid, kuid vähestes olemasolevatest, saab planeerida mitme sihtkohaga lennureisi ning nende probleemiks on, et sihtkohas viibimise perioodi peab sisestama manuaalselt. Paljud otsingusüsteemid kombineerivad mitme sihtkohaga lennureisi koostamisel reisi vaid ühte lennufirmat kasutades, mille tõttu jääb sobiv marsruut üldse leidmata, sest antud lennufirma sihtkohad on piiratud.

Käesoleva bakalaureusetöö eesmärgiks on luua automatiseeritud mitme sihtkohaga lennureisi otsingusüsteem, millega saab aega kokku hoides leida kõige otstarbekam lennureis, mis kasutab selleks erinevaid lennufirmasid. Loodud veebirakenduses on võimalik sisestada mitu sihtkohta, kuhu kasutaja ühe reisi jooksul reisida soovib, ning perioodi, mis vahemikus see reis toimuda võiks. Sisestatud andmete põhjal loob rakendus kõik võimalikud reisirakendused.

Töö teises peatükis analüüsib autor olemasolevaid lahendusi, mis võimaldavad otsida lennupileteid. Töö kolmandas ja neljandas peatükis annab autor ülevaate veebirakenduste arendamise aspektidest ning veebirakenduses kasutatud tehnoloogiatest. Töö viiendas peatükis kirjeldatakse töö käigus valminud rakenduse arenduskäiku. Kuues peatükk analüüsib loodud veebirakendust.

2 Veebirakendused

Veebirakendus on mistahes klient-server tarkvara rakendus, mille puhul kliendiks on veebibrauser. Kõikidele interneti kasutajatele on veebirakendused kõige lihtsamini kätte saadavad ning seetõttu asuvad paljud teenused just veebirakendustes. Veebirakendusi arendatakse enamasti raamistike abil, kus raamistik on valmis lahendus, mida saab korduvate ülesannete puhul kasutada. Tavaliselt kasutatakse ühes veebirakenduses mitut erinevat raamistikku, mis toetavad erinevaid valdkondi [1].

2.1 Veebirakenduste eelised ja puudused

Veebirakendused ei nõua konkreetset tüüpi arvutit või operatsioonisüsteemi, sest rakendus jookseb veebibrauseris [1]. See tähendab, et veebirakendused ei sõltu mitte millestki muust kui vaid veebibrauserist. Samuti pole vaja veebirakendusi alla laadida, et neid kasutada. Veebirakenduste eeliseks võib nimetada ka seda, et neid on lihtne uuendada ja täiustada, sest uuendusi on vaja üles laadida vaid veebiserverisse ning kohe peale seda saavad rakenduste kasutajad neid uuendatud versioone näha.

Veebirakendusi on lihtne arendada, sest vabavaras on olemas väga palju tööriistu ning raamistikke nende arendamiseks. Need mitte ainult ei kiirenda arendamist vaid ka aitavad ka arendamise hinda sellega madalana hoida [23].

Veebirakendused jooksevad serveris koos andmetega, millele on vaja juurdepääsu. Seda saab vaadelda kui turvariski, sest kõik andmed, mida veebirakendus kasutab, hoitakse keskses kohas ning igasugune oht julgeolekule tähendab, et kõik andmed on ohustatud [23].

Kuna kõik andmed ja loogika veebirakendusele hoitakse serveris, siis kõikidel kasutajatel on vaja veebirakenduse kasutamiseks internetiühendust. Kui ühendust pole, siis muutub veebirakendus täiesti kasutamatuks. Kuna veebirakendus on kättesaadav vaid interneti kaudu, siis internetiühenduse kiirus saab ka üheks piiravaks teguriks, sest

suurte koguste andmete saatmine üle aeglase internetiühenduse võib olla väga aeganõudev [23].

2.2 Veebirakenduste turvalisus

Arvestades, et kõik kes kasutavad interneti, kasutavad ka veebirakendusi, siis veebirakenduste turvalisus on peamine, millele tähelepanu pöörata. Igaüks on vähemalt korra elus kokku puutunud mingisuguse küberrünnakuga ning seepärast on oluline rõhuda veebirakenduste turvalisusele.

Veebirakenduse turvalisuse all defineeritakse meetodeid ja põhimõtteid, et vältida ja tuvastada turvaohete [9]. Veebirakendus on turvaline, kui selle rakenduse infole ning andmetele pole võimalik ligi pääseda, sest veebirakendused võivad sisaldada konfidentsiaalseid andmeid.

Põhilised veebirakenduste turvakomponendid [1]:

- Autentimine – selleks, et saada juurdepääs veebirakenduse kaitstud osadele, peab kasutaja end autentima.
- Volitused – selleks, et kõikidel kasutajatel puuduks ligipääs kohtadesse, mis on neile keelatud.
- Sisestuse ja väljastuse valideerimine – sisestatud ja väljastatud andmeid tuleb kontrollida ja filtreerida, et vältida kahjulike andmete töötlemist.
- Seansihaldus – kuna internetiprotokoll HTTP ei võimalda päringuid siduda konkreetse kasutajaga, siis tuleb seostamiseks kasutada veebirakenduse seanssihaldust.
- Tõrkekäsitlus – tekkinud tõrke korral peab seda käsitleda selliselt, et rakenduse andmete turve säiliks.
- Logimine - selleks, et välja selgitada tagantjärele aset leidnud tegevusi ja võimalike intsidentide tagamaid.

Turvalisuse tagamiseks ei piisa vaid veebirakenduse turvalisusest. Ohutus tugineb nii võrgu, operatsioonisüsteemi kui ka veebiserveri turvalisusele, nagu ka terve see platvorm seda teeb [9].

3 Olemasolevad lahendused

Lennupiletite otsingusüsteeme on mitmeid. Suuremad otsingusüsteemid koguvad andmeid teistelt veebilehtedelt üle interneti, kus on välja toodud lennupiletid ning panevad tulemused kokku ühte kohta ilma, et kasutaja peaks otsima lennupileteid igalt lennufirmalt eraldi. Väiksemad lennupiletite otsingusüsteemid kasutavad API-si, kus nende eest on suurem otsing ära tehtud. Seda varianti kasutatakse ka antud lõputöös.

Momondo on samuti meta otsingumootor. See loodi 2006. aasta septembris Taanis. Kõigepealt pakkus momondo vaid lennupiletite otsingut. Nüüdseks pakub Momondo ka hotelli otsingu ning autorendi võimalust [24]. Selle eesmärgiks on pakkuda kasutajatele sõltumatut lennupiletite otsingusüsteemi, mis pakub täielikku hindade läbipaistvust [25]. Momondo pakub lisaks hinna ja kiiruse sorteerimisele ka parima lennureisi valikut, mis leiab reisi, kus on arvesse võetud nii eelarvet kui mugavust [28].

Kayak on meta otsingumootor, mis otsib teistelt lennupileti otsingu veebilehtedelt, lennupiletite reisibüroodelt, hotelli ja auto rendi veebilehtedelt informatsiooni kasutajale, et kasutaja saaks teha parima otsuse endale sobiva reisi leidmiseks [29]. Kayaki teeb eriliseks see, et see pakub palju erinevaid viise tulemuste filtreerimiseks, et leida parim reis [28].

Skyscanner on üks populaarsemaid meta otsingumootoreid, kus saab otsida lennupileteid. Peale lennupiletite saab selle kaudu otsida ka hotelle ning ka autorendi võimalust. Skyscanner on loodud 2003 aastal Gareth Williams, Bonamy Grimesi ja Barry Smithi poolt eesmärgiga lihtsustada odavate lennupiletite leidmist kiiremalt [18]. Skyscanneri teeb eriliseks asjaolu, et sellel on võrdlemisi lihtne kasutajaliides [28]. See annab võimaluse leida kiiresti endale sobiv lennureis.

Kiwi.com (endise nimega Skypicker) on online reisibüroo. Kiwi.com kogub reaajas lennuandemeid lugematutelt lennufirmadelt ja lennu andmekoondajatelt ning salvestab tulemused andmebaasi. See unikaalne lennu otsingu algoritm, võimaldab kombineerida lende nende lennufirmade vahel, mis tavaliselt ei paku ühtset lennuplaani. Selle tulemusena on võimalik märkimisväärselt raha säästa [17].

Kiwi.com pakub ka lennupiletite otsingu API-t Skyscanner, mida antud töös kasutatakse.

3.1 Olemasolevate lahenduste analüüs

Võrdlusesse on võetud neli lennupiletite otsingumootorit (Tabel 1). Kolm neist on valitud veebilehe Techlicious artikli „The Best Flight Search Services“ põhjal, kus on välja toodud parimad lennupiletite otsingu veebilehed, ning autor on valitud neist kolm, mida ise varem kasutanud on. Viimane (Kiwi.com) on võetud võrdlusesse, sest autor kasutab käesoleva lõputöö koostamisel just selle lennuotsingumootori API-t.

Kõik võrdlusesse võetud lennupiletite otsingumootorid võimaldavad otsida nii ühe suuna, edasi-tagasi kui ka mitme sihtkohaga reise. Viimase valiku puhul peab reisija teadma, mis järjekorras ning millistel kuupäevadel soovib ta sihtkohti läbida. See probleem ajendas autorit kirjutama antud lõputööd.

Kõik otsingumootorid pakuvad ka võimalust leida ükskõik millise sihtpunkti lennupileteid ning järjestatakse tulemused hinna järgi odavamast kallimani. Momondogi pakub sama võimalust, kuid ei näita hinda enne kui kõikide sihtkohtade seast üks on valitud.

API-t pakuvad kaks võrdlusesse toodud lennureisi otsingumootorit: Skyscanner ja Kiwi.com (edaspidi Skypicker API). Skyscanneri API on vabavaras saadaval kogu ulatuses, kuid Skypicker API on vabavaras saadavad vaid test versioonis. Autor kasutas Skypicker API-t, sest Skyscanneri API oli veebirakenduse arendamise alustamise ajal teadmata ajaks hoolduses.

Nimetus	Ühe suuna reis	Edasi-tagasi reis	Mitmesihtkohaga reis	Kõikjale otsing korraga	API
Skyscanner	Jah	Jah	Jah	Jah	Jah
Kayak	Jah	Jah	Jah	Jah	Ei
Momondo	Jah	Jah	Jah	Võimaldab, kuid ei näita kohe hinda	Ei
Kiwi.com	Jah	Jah	jah	Jah	jah

Tabel 1. Olevasolevate lahenduste võrdlus

4 Reisiplaneerimisrakenduse arendus

4.1 Arenduskäik

Käesoleva töö autor alustas arendust sobiva lennupileti otsingu API otsimisega. Mitmete katsetuste tulemusena, otsustas autor valida Skypicker API. Seejärel selgitas ta välja, milline on kõige parem viis PHP-s teha HTTP päringuid.

Veebirakenduse aredamist alustas autor sellest, et tegi eraldi kaustad JavaScript-i, HTML-i ja CSS-i jaoks, et rakenduse erinevad osad oleksid eraldatud ning kiiresti leitavad. PHP klasside nimed grupeeris antud rakenduse vaadete järgi, kus nime esimene osa on pandud vaate järgi ning teine osa kirjeldab, mida selles klassis tehakse. Kohe alguses ühendas autor oma projekti ära ka versioonihaldustarkvaraga GIT, et arendamisel tekkivate probleemide käigus oleks olemas töötav lahendus. Edasine arendus käis erinevate funktsionaalsuste kaupa järjestades need keerulisemast lihtsamani.

Esmalt sai valmis arendatud mitme sihtkohaga lennupiletite otsingu vaade ning selle loogika. Seejärel täiendas autor otsingu funktsionaalsust, ning läbi katsetuste leidis kiireima viisi asünkroonsete päringute tegemiseks. Samuti pidi silmas pidama seda, et algoritm ei töötaks liiga palju mõttetuid andmeid. Peale mitme sihtkohaga lennupiletite otsingu arendust valmis ühe sihtkohaga lennupiletite otsingu vaade ning selle funktsionaalsus. Viimasena viimistles autor veebirakenduse kasutajaliidest.

Arenduse käigus pidas selle looja kõige enam silmas järgida, et kood oleks võimalikult hästi hallatav. Selleks üritas autor võimalikult palju oma koodi tükeldada, et kogu kood ei oleks ühes funktsioonis või ühes klassis. Samuti üritas autor võimalikult arusaadavaid funktsiooni nimesid panna, mis oleks iga meetodi tegevusega vastavuses.

4.2 Arendamisel kasutatud tehnoloogiad

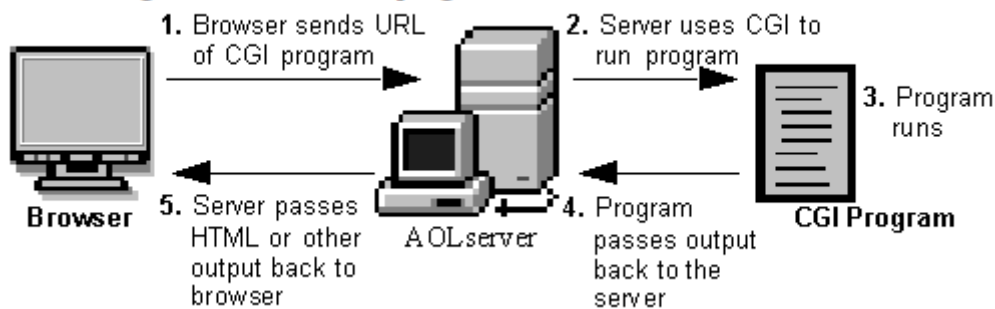
Lõputöö arendamisel kasutati CGI tehnoloogia komplekti, kus PHP-d kasutati rakenduse ehitamiseks, Skypickeri API-t lennupiletite otsinguks, jQueryt kasutajaliidese realiseerimiseks, Bootstrapi kasutajaliidese ülesehitamiseks ning funktsionaalsuse ja kasutajaliidese lahku lõõmiseks Twig-i.

4.2.1 CGI

CGI (*Common Gateway Interface* - ühinelüüsiliides) on meetod, millega saab ühendada veebirakenduse veebiserveriga [10]. See tähendab, et andmed mida kasutaja sisestab veebirakendusse omakorda antakse edasi CGI rakendusele, mis omakorda dünaamiliselt genereerib veebisisu, mida brauserile tagastada.

CGI-d saab kasutada ükskõik milline programm, mis jookseb veebiserveris ning aktsepteerib käsurea andmeid [10]. Lahti seletatult töötab CGI nii, et brauser saadab URL-i serverile, et kasutada CGI-d, mis käivitab programmi. Programm tagastab, mida sellelt päriti läbi CGI serverile ning server saadab vastuse edasi brauserile. CGI on nagu värv serveri ja programmi vahel, mida on kirjutatud.

Here is a diagram of how a CGI program runs:



Joonis 1. CGI programmi töötamise joonis

4.2.2 Skypicker API ja Google Maps API

API (*Application Programming Interface* - rakendusliides) on kogum rutiinidest, protokollidest ja vahenditest rakenduste ehitamiseks. API kirjeldab, kuidas tuleks suhelda tarkvarakomponentidega [19].

Google Maps API annab arendajatel lisada Google Maps kaardi veebilehele või isegi saada andmeid selle kaardilt [20]. Antud bakalaureuse töös kasutatakse JavaScripti põhiste Google Maps API-t kaardi kuvamiseks (Joonis 2).

```
MAP.initialize = function() {
    var tallinn = { lat: 59.42831578404258, lng: 24.774169921875 };
    var map = new google.maps.Map(document.getElementById('map'), {
        zoom: 4,
        center: tallinn
    });

    google.maps.event.addListener(map, 'click', function(event) {
        MAP.addMarker(event.latLng, map, event);
    });
};

google.maps.event.addDomListener(window, 'load', MAP.initialize);
```

Joonis 2. JavaScript Google Mapsi kaardi loomiseks

Skypicker on lennupiletite otsingu rakendusliides (API). Skypicker API kasutab IATA (*International Air Transport Association* – Rahvusvaheline Lennutranspordi Ühendus) lennujaamade ja ISO (*International Organization for Standardization* – Rahvusvaheline Standardiorganisatsioon) riikide koode lennupiletite otsinguks [14]. Antud lõputöös kasutatakse /places ja /flights API väljakutseid, kus /places väljakutsega, saab leida sihtkohad ning /flights väljakutsega saab otsida sobivate parameetritega lennureise.

Autor valis antud lõputöö raames Skypicker API, sest selle testversioon on vabavaras saadaval ning seda on lihtne kasutada, mis seetõttu arendamise kiiremaks muudab.

4.2.3 PHP

Rakenduse ehitamiseks kasutati PHP-d. PHP on laialdaselt kasutatav avatud lähtekoodiga skriptikeel, mis kõige enam on mõeldud veebi arenduseks [2]. Seda saab kasutada kõikides suuremates operatsioonisüsteemides (Linux, Microsoft Windows, Mac OS X). Kuna PHP põhiliselt fookuseerib serveripoolsele skriptimisele, siis saab sellega teha andmebaasi päringuid, genereerida dünaamilist veebilehe sisu või isegi saata küpsiseid [3].

Antud töös kasutati HTTP päringute tegemiseks HTTP library-t Requests, mis on kirjutatud PHP-s (Joonis 3). Requests pole ainuke HTTP library PHP-s, kuid eriliseks teeb selle see, et see on mõeldud kasutama erinevate süsteemidega, sest see kasutab

cURL-i kui see on saadavad, teistel juhtudel kasutab see socketeid [12]. See muudab ka päringute tegemise kiiremaks.

```
public function getPlaces( $search )
{
    $url=$this->URL.'/places?term='.$search.'&locale=en&v=3';
    $headers=array(
        'Accept' => 'application/json'
    );
    $response=Requests::get($url,$headers);
    return $response->body;
}
```

Joonis 3. PHP Requests päring sihtkohaotsimiseks

cURL on viis kommunikeerida erinevat tüüpi serverite ja protokollidega. Teisisõnu on cURL vahend, millega saab teha HTTP päringuid PHP-s, mis on üks olulisemaid osi antud lõputöös. PHP pakub libcurl library-t, mis on tehtud Daniel Stenbergi poolt. See toetab HTTP ja HTTPS protokolle, mida antud töös ka kasutatakse [22].

Socket on kahe-suunalise siduühenduse link kahe võrgus töötava programmi vahel. Socketid käituvad nagu faili deskriptorid, sest käsud read() ja write() töötavad socketitega samamoodi nagu failidega [13]. Antud töös socketeid ei kasutata, sest socketeid kasutatakse pideva andmevahetuse jaoks.

4.2.4 JavaScript ja jQuery

JavaScript-i tuntakse kõige enam kui veebilehtede scriptimiskeelt, kuid seda kasutatakse ka paljudes teistes platvormides. JavaScript töötab kliendi poolel veebis ning selle ülesanne on kontrollida veebilehe käitumist [16].

jQuery on JavaScripti Library, mis on loodud John Resigi poolt 2006. aastal. Selle eesmärk on lihtsustada koodi kirjutades vähem koodi [5]. See tähendab, et on võimalik kasutada JavaScripti väga väikesel tasemel nii, et saab kirjutada palju vähem koodi. See teeb JavaScripti kasutamise palju lihtsamaks.

Peamine põhjus, miks autor kasutab antud töös jQuery, sest sellega on võimalik teha AJAX (*Asynchronous Javascript And XML* – asünkroonne Javascript ja XML) päringuid. AJAX ei ole programmeerimiskeel, vaid see on kombinatsioon andmete serverist pärimisest ning nende andmete kasutamisest. See võimaldab veebilehtedel uuendada

asünkroonselt andmeid vahetades neid serveriga [6]. See tähendab, et on võimalik uuendada veebilehte, seda uuesti laadimata.

4.2.5 Bootstrap

Veebirakenduse kasutajaliides on üles ehitatud kasutajaliidese raamistiku abil Bootstrap. Bootstrap on üks populaarsemaid HTML, CSS ja JavaScript raamistikke, millega saab arendada skaleeruvaid veebirakendusi [11].

Bootstrapis on sisseehitatud skaleeruva disaini tugi, mis teeb lihtsamaks veebilehtede loomist erinevatele nutiseadmetele. See on oluline, sest rohkem kui pool interneti liiklusest käib läbi just nutiseadmete.

HTML (*Hypertext Markup Language* – märgistuskeel) on standartne märgistuskeel, et luua veebilehti ja veebirakendusi. See koosneb mitmetest väikestest koodi osadest tekstifailis, mida nimetatakse tag-deks. Need tag-id on nagu ehitusblokid, mille tulemusena välja tuleb veebileht.

CSS (*Cascading Style Sheets* – astmelised stiililehed) on keel veebilehtede kujundamiseks, mis täpsemalt öeldes tähendab, et tegemist on keelega, mis kirjeldab kujundust. CSS võimaldab muuta HTML elementide värvust, kuju, tausta, käitumist, põhimõtteliselt kõike. Selle kasutamist soovitab ka W3C (*World Wide Web Consortium* -) – organisatsioon, mille eesmärgiks on interneti arendamine ning veebistandardite väljatöötlemine [27].

4.2.6 Twig

Twig on mallimootor, mis on mõeldud lahutamaks rakenduse funktsionaalsuse ja kasutajaliidese [15]. See muudab koodi hallatavamaks, sest erinevad osad on lahku löödud. Mallid on kirjutatud HTML-is.

4.3 Arendamisel kasutatud tööriistad

4.3.1 PhpStorm

PhpStorm on IDE (*integrated development environment* – sisseehitatud programmeerimiskeskond) platvorm PHP kirjutamiseks ning selle on ehitanud IntelliJ IDEA. See pakub automaatset täitmist, lennult vigade tuvastamist ning koodi

refaktoreerimist. Samuti pakub PhpStorm võimsaid sisseehitatud tööriistu silumiseks ning testimiseks. PhpStormis on sisse ehitatud HTML, CSS ja JavaScripti tekstitoimetajad, mille jaoks pole vaja midagi konfigureerida [20]. Eelnimetatud teksitoimetajaid kasutatakse ka antud lõputöös.

4.3.2 GIT

Git on versioontarkvara, mis loodi algselt Linud Torvaldi poolt Linuxi tuuma arendamiseks [7]. Git-i kasutatakse failide salvestamiseks, et aja jooksul oleks võimalik konkreetset versiooni failist tagasi kutsuda, olles põhjuseks, mingisuguse probleemi tekkimine failis või isegi selle kadumine.

5 Valminud rakendus

Valminud rakenduses on võimalik otsida automatiseeritud mitme sihtkohaga lennureisi. Samuti on võimalik otsida ühe sihtkohaga lennureisi nii edasi-tagasi variandis kui ka ühe otsa variandis. Kogu rakendus on valminud ühe lehe formaadis, kus vaadete vahel on võimalik navigeerida vahekaartide abil.

Kasutaja geosukoht on rakenduses pandud automaatselt reisi alguspunktiks ning mitme sihtkoha otsingu puhul ka lõpp-punktiks. Selleks on kasutatud API-d millega on võimalik kasutaja IP aadressi järgi saada tema asukoht.

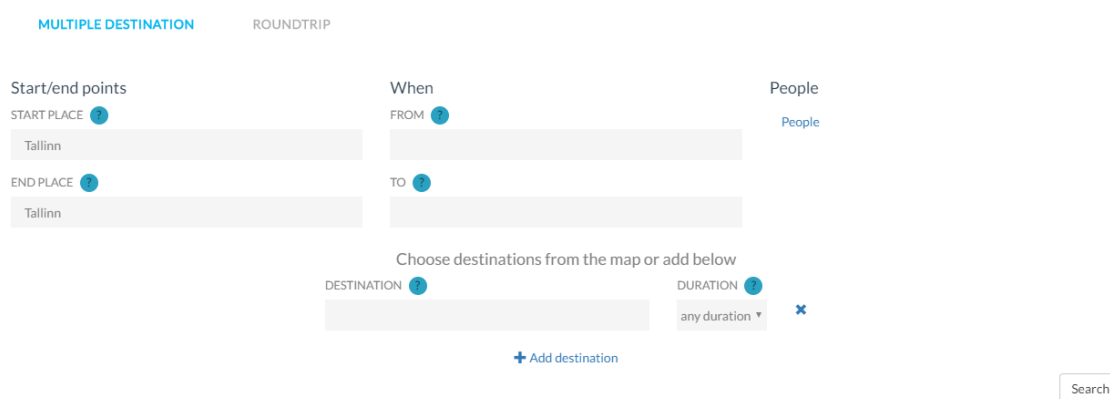
Nii mitme sihtkohaga kui ka ühe sihtkohaga lennureisi otsingu vaates saab peale manuaalse otsimise ka valida sihtkohti Google Maps-i kaardi kaudu, mis asub otsingubloki üleval (Joonis 4). Valides kaardilt sihtkoha ning seejärel vajutades sellele lisatakse sinna lipuke ning hüppab lahti hüpikaken, kust tulevad ette võimalikud sihtkohad, mis on selle punkti läheduses. Valides sihtkoha tuleb peale seda valida, milleks see mõeldud on, kas algupunkt, lõpp-punkt või mitme sihtkohaga otsingu puhul sihtkoht, kuhu reisi käigus soovitakse reisida.



Joonis 4. Google Maps kaart sihtkoha märgiga

5.1 Mitme sihtkohaga lennureisi otsingu vaade

Mitme sihtkohaga lennureisi otsingu vaates on võimalik otsida mitme sihtkohaga reisi. Seal on kuvatud otsingu blokk, mis koosneb kahest osast (Joonis 5). Esimeses osas saab sisestada reisi algus- ning lõpp-punkti ning reisiga seotud üldiseid andmeid: reisi periood, inimeste arv. Teises osas saab lisada sihtkohad, kuhu oleks soov reisida. Samuti vajadusel saab seal valida, kui mitu päeva soovitakse teatud sihtkohas viibida. Vajutades nuppu „Search“ hakkab loodud algoritm otsima sisestatud parameetrite järgi reisi.



The screenshot shows a flight search form with two tabs: "MULTIPLE DESTINATION" (selected) and "ROUNDRIP". The form is divided into three main sections: "Start/end points", "When", and "People".

- Start/end points:** Contains "START PLACE" and "END PLACE" fields, both with "Tallinn" entered.
- When:** Contains "FROM" and "TO" fields, both empty.
- People:** Contains a "People" label and a "People" input field.

Below these sections, there is a section for destinations with the text "Choose destinations from the map or add below". It includes a "DESTINATION" input field, a "DURATION" dropdown menu set to "any duration", and a "+ Add destination" button. A "Search" button is located at the bottom right of the form.

Joonis 5. Mitme sihtkohaga otsingu blokk

Mitme sihtkohaga lennureisi otsingu tulemused kuvatakse otsingu bloki alla. Neid saab sortida nii hinna järgi, mis on vaikimisi seadistatud, kui ka lennureisi kestvuse järgi, kui kaua terve reisi jooksul inimene lennukis peab viibima. Samuti on võimalik filtreerida tulemusi hinnavahemiku järgi. Tulemuste sortimine ja filtreerimine käib eessüsteemi poole peal. See tähendab, et kogu HTML on põhiprogrammi poolt genereeritud ning uusi päringuid serverile enam ei tehta.

Otsingu tulemused on jaotatud teekondade järgi blokkidesse. Blokil on märgitud teekond, odavaim reis ning kõige lühema lennuajaga reis. Vajutades blokile avanevad kõik sellele teekonnale vastavad tulemused. Seal on kuvatud olulisim info lennupiletite kohta: hind, aeg, lennufirma.


Esmane tegevus, mis „Search“ nuppu vajutades tööle hakkab on lennureisi kõikide sihtkohtade järjestuste leidmine. Kõik võimalikud järjestused leitakse kasutades

matemaatilist permutatsioonide valemit, mis tähendab, et leitakse kõikide hulga elementidest jada, kus iga element esineb täpselt ühe korra. Tagastatud tulemuste põhjal hakatakse asünktoonselt otsima lennupileteid.

Asünktoonse otsingu kasutamine, mis antud juhul valitud on, muudab veebirakenduse kiiremaks – kõik päringud tehakse nii, et nad üksteisest ei sõltuks. Samuti ei koorma see variant väga palju programmi.

Enne API päringute tegemist leitakse kõik võimalikud perioodid, kui kaua antud sihtkohas viibitakse. Juhul kui kasutaja on märkinud kindla sihtkohas viibimisperioodi, siis seda arvestatakse ning kui pole märgitud, siis algoritm ise pakub variante. Kuna lõputöö eesmärk on leida otstarbekaid reise, siis on elimineeritud ebaloogilised perioodid. See tähendab, et arvesse pole võetud ebareaalseid perioode, kus reisi kogu kestvus on pikk näiteks 30 päeva ja ühe sihtkoha viibimise periood on väga väike näiteks 1 päev.

Peale kõik võimalike perioodide leidmist tehakse HTTP päring Skypicker API pihta ning genereeritakse HTML saadud tulemuste alusel (Joonis 6). Kõik API päringud tehakse üksteise järgi. See on seepärast oluline, et saada lennupileteid sellises järjekorras, kuidas reis toimuma hakkab. Vastasel juhul oleks raske hiljem tulemusi kokku sobitada.

	Tallinn (TLL)Barcelona (BCN) Stops: 1	25.05.2017 23:35-12:10	
	Barcelona (BCN)Malaga (AGP) Stops: 1	26.05.2017 18:20-11:20	448€
	Malaga (AGP)Berlin (SXF) Stops: 1	27.05.2017 14:55-23:40	
	Berlin (TXL)Tallinn (TLL)	01.06.2017 10:00-12:50	

Joonis 6. Mitme sihtkohaga otsingu tulemus

5.2 Ühe sihtkohaga lennureisi otsingu vaade

Ühe sihtkohaga lennureisi otsingu vaates (Joonis 6) saab otsida reisi kindla sihtkoha järgi kui ka leida kõige odavamad reisi mis tahes sihtkohta. Samuti on seal võimalik leida edasi-tagasi lennuga reisi kui ka ühe otsa piletiga reisi. Peale algus- ja lõpp-punkti ning kuupäevade valimist saab ka soovi korral lisada teisi parameetreid reisi otsimiseks: inimeste arvu, nagu ka mitme sihtkohaga reisiotsingu puhul, ning valida, kas soovitakse ainult otselende valitud sihtkohta. Kohustuslik on alati valida lähtepunkt ning reisi algustamise kuupäev.

The screenshot shows a flight search form with the following elements:

- Buttons: **MULTIPLE DESTINATION** (grey), **ROUNDRIP** (blue)
- Section: **Start/end points**
 - FROM: Tallinn
 - TO: (empty)
- Section: **When**
 - DEPART: (empty)
 - RETURN: (empty)
- Section: **How**
 - People: (empty)
 - ONLY DIRECT FLIGHTS: (checkbox)
- Search button: Search

Joonis 7. Ühe sihtkohaga lennureisi otsingu blokk

Ühe sihtkohaga otsingu vaates on võimalik otsida ka lennureisi sihtkohta lisamata. Sellisel juhul leitakse kõik võimalike sihtkohtadega lennureisid ette antud sihtpunktist. Nii nagu ka mitme sihtkohaga otsingu puhulgi kuvatakse kõik tulemused otsingubloki alla.

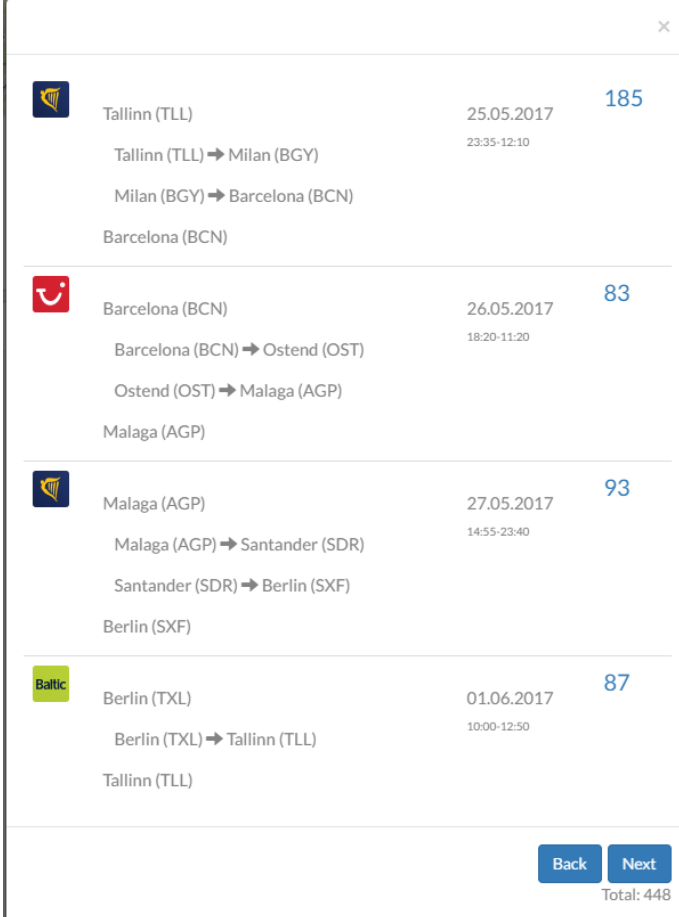
Tulemusi on võimalik sorteerida hinna ning lennuaja järgi. Hinna järgi sorteerimine on vaikimisi seadistatud.


5.3 Leitud reisi detailvaade

Antud töös on kuvatud reisi detailvaade hüpikakna kujul. Sinna on kuvatud info lennupiletitest. Peale lennupileti hinna on seal kuvatud ka muu lennuga seonduv oluline informatsioon.

Detailvaatest on olemas ka link lennupileti ostmiseks, mis suunab Kiwi.com lehele, kus on võimalik lennupileti eest tasuda. Mitme sihtkohaga otsingu puhul on antud iga lennu jaoks eraldi lennupileti link. See on seepärast, et oleks võimalik reis kokku panna

kasutades erinevaid lennufirmasid, vastasel juhul võib juhtuda, et puudub ette antud reis. Kui autor oleks kasutanud lennupiletite otsinguks Skypickeri poolt pakutavat /flights_multi väljakutset, siis poleks kasutaja poolt määratud parameetritega mõningatel juhtudel ühtegi tulemust ilmunud.



	Tallinn (TLL) Tallinn (TLL) → Milan (BGY) Milan (BGY) → Barcelona (BCN) Barcelona (BCN)	25.05.2017 23:35-12:10	185
	Barcelona (BCN) Barcelona (BCN) → Ostend (OST) Ostend (OST) → Malaga (AGP) Malaga (AGP)	26.05.2017 18:20-11:20	83
	Malaga (AGP) Malaga (AGP) → Santander (SDR) Santander (SDR) → Berlin (SXF) Berlin (SXF)	27.05.2017 14:55-23:40	93
	Berlin (TXL) Berlin (TXL) → Tallinn (TLL) Tallinn (TLL)	01.06.2017 10:00-12:50	87

Back Next
Total: 448

Joonis 8. Hüpinkakna vaade

Hüpinkakende vahel on võimalik liikuda edasi („Next“) ja tagasi („Back“) nuppudega, mille tulemusena kuvatakse ka järjestuses järgmine või eelmine tulemus. See võimaldab kiiresti liikuda detailvaadete vahel ilma, et tuleks hüpinkaken sulgeda. Liikumise nupud arvestavad sellega mismoodi on tulemused sorditud, et ei tekiks arusaamatusi, mis loogika selle taga on. Mitme sihtkohaga otsingu puhul on võimalik liikuda hüpinkakende puhul vaid valitud bloki piires.

6 Reisiplaneerimisrakenduse analüüs

Lõputöö funktsionaalsust ja kasutajamugavust lasi autor testida kasutajatel, kes igapäevaselt ei puudu kokku arendamisega ning lennupiletite otsimisega, et uurida kas rakendus töötab planeeritud eesmärgiga ning on arusaadav. Testimine teiste kasutajate poolt peale autori on oluline, sest tihti ei oska autor kõiki vigu märgata olles kogu arendusprotsessi juures.

Kasutajad tõid välja, et veebirakendus pole piisavalt arusaadav. Selle lahendamiseks lisas autor erinevate tegevuste juurde küsimärgiga nupud, et juhul kui tekib küsimus, siis saab sellele peale vajutades teada, mida tegema peab. Samuti toodi välja, et väikese ekraaniga nutiseadmetel pole piisavalt mugav antud rakendust kasutada.

Arendamise käigus pidi veel silmas pidama, et veebirakendust oleks mugav kasutada ka erinevates veebibrauserites ning nutiseadmetes. Testid viidi läbi veebibrauserites Chrome, IE/Edge, Firefox ja Safari. Samuti viidi testil läbi nii Androidi kui ka IOS-i veebibrauserites. Oluline oli, et kasutajaliides ning funktsioonid töötaksid igas eelnevalt nimetatud veebibrauseris sama moodi. Nutiseadmete puhul tuli veel jälgida, et kasutajaliides kohanduks väiksema ekraaniga.

Kõige enam raskusi oli IE/Edge veebibrauseriga, sest veebirakenduse üks komponente, kaardilt sihtkoha leidmine ei töötanud nii nagu oli soovitud. Samuti pidi muutma Edge kujunduslikke elemente, sest need erinesid teistest brauseritest.

Kuna Kiwi.com, mis pakub lennupiletite otsinguks vabavaras Skypicker API, on online reisibüroo, siis võivad lennupiletite hinnad erineda lennufirmade poolt pakutavate pooldest. See on seetõttu, et reisibürood võtavad vahendustasu.

Töö nõrgaks küljeks võib nimetada, et suurte päringute tegemisel läheb tulemuste saamisega kaua aega. Suurte päringute all peab autor silmas variant, kus mitme sihtkohaga reisi otsinguks valitakse üle viie sihtkoha. Aja kokkuhoiu mõttes on autor pannud päringutele aja piirangu kui kaua rakendus lennureise otsib. Sellest tulevana

võib puududa kõige otstarbekam reis, sest mõningate reise tulemuste otsing jäi poolikuks.

7 Kokkuvõte

Käesolevas bakalaureuse töös analüüsiti erinevaid lennupiletite otsingu mootoreid. Neid uurides selgus, et paljud pakuvad mitmesihtkohaga reisi otsimist, kuid ei paku võimalust otsida mitme sihtkohaga reisi automatiseeritult. Mõni kombineerib reisi vaid ühte lennufirmat kasutades, mis tõttu võib selline marsruut üldse puududa, sest lennufirma sihtkohad on piiratud.

Töö tulemusena valmiski veebirakendus, millega on võimalik otsida automatiseeritud mitme sihtkohaga lennureisi. Samuti valmis ka võimalus otsida vaid ühe sihtkohaga lennureisi nii edasi-tagasi variandis kui otse lennu variandis.

Automatiseeritud mitme sihtkohaga lennureisi otsingusüsteem aitab kasutajal leida kiiremini mitme sihtkohaga lennureisi kui pole oluline sihtkohtade järjestus. Samuti kombineerib rakendus reisi kasutades erinevaid lennufirmasid.

Bakalaureusetöö eesmärk sai täidetud. Valmis realselt töötav veebirakendus, mida saab kasutada nii arvutis kui ka nutiseadmes.

Kasutatud kirjandus

- [1] Veebirakendus [WWW] <https://www.ria.ee/public/ISKE/Veebirakendus.pdf> (05.05.2017)
- [2] What is PHP? [WWW] <http://php.net/manual/en/intro-whatis.php> (05.05.2017)
- [3] What can PHP do? [WWW] <http://php.net/manual/en/intro-whatcando.php> (05.05.2017)
- [4] What is CGI? [WWW] <http://www.cgi101.com/book/intro.html> (05.05.2017)
- [5] What is jQuery? [WWW] <https://www.tutorialspoint.com/jquery/jquery-overview.html> (05.05.2017)
- [6] AJAX Introduction [WWW] https://www.w3schools.com/xml/ajax_intro.asp (18.05.2017)
- [7] Getting started – About Version Control [WWW] <https://git-scm.com/book/en/v2/Getting-Started-About-Version-Control> (18.05.2017)
- [8] What is Skyscanner? [WWW] <https://www.skyscanner.net/news/tools/what-is-skyscanner/> (20.05.2017)
- [9] Rojas, E. C. R (2013) - [online] <http://hh.diva-portal.org/smash/get/diva2:610574/FULLTEXT01.pdf> (10.05.2017)
- [10] Hayes, F. Common Gateway Interface – [online] (10.05.2017)
- [11] Bootstrap [WWW] <http://getbootstrap.com/> (10.05.2017)
- [12] Requests for PHP [WWW] <http://requests.ryanmccue.info/docs/goals.html> (17.05.2017)
- [13] What is a Socket? [WWW] <http://journals.ecs.soton.ac.uk/java/tutorial/networking/sockets/definition.html> (17.05.2017)
- [14] Skypicker.com API [WWW] <http://docs.skypickerpublicapi.apiary.io/#introduction/how-it-works> (17.05.2017)
- [15] Template Engines [WWW] <https://www.simple-is-better.org/template/> (17.05.2017)
- [16] About JavaScript [WWW] https://developer.mozilla.org/en-US/docs/Web/JavaScript/About_JavaScript (17.05.2017)
- [17] About Kiwi.com [WWW] <https://www.kiwi.com/en/content/about> (20.05.2017)
- [18] What is SkyScanner? [WWW] <https://www.skyscanner.net/news/tools/what-is-skyscanner/> (20.05.2017)
- [19] Beal, V. API – application program interface [WWW] <http://www.webopedia.com/TERM/A/API.html> (20.05.2017)
- [20] About PhpStorm [WWW] <https://www.componentsource.com/product/phpstorm/about> (21.05.2017)
- [21] What are the Google Map APIs? [WWW] <https://developers.google.com/maps/faq#whatis> (21.05.2017)
- [22] Introduction to cURL [WWW] <http://php.net/manual/en/intro.curl.php> (21.05.2017)
- [23] What is Web Application Software? (2016) [WWW] <https://www.bespokesoftwaredevelopment.com/blog/what-is-web-application-software/> (21.05.2017)

- [24] Momondo [WWW] <http://www.thegeniusworks.com/gamechanger/momondo/> (21.05.2017)
- [25] About Momondo [WWW] <http://www.momondo.ee/content/about/> (21.05.2017)
- [27] CSS/Sissejuhatus [WWW] <https://et.wikibooks.org/wiki/CSS/Sissejuhatus> (21.05.2017)
- [28] Stokes, N (2017) Thw Best Flight Search Services [WWW] <http://www.techlicious.com/tip/the-best-flight-search-services/> (21.05.2017)
- [29] About Kayak [WWW] <https://www.kayak.com/about> (21.05.2017)