

IMPLEMENTACIJA PROGRAMSKOG RJEŠENJA ZA PODRŠKU UPRAVLJANJA TENISKIM DOGAĐAJIMA

Klepac, Mijo

Undergraduate thesis / Završni rad

2023

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Algebra University College / Visoko učilište Algebra**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:225:852063>

Rights / Prava: [In copyright](#) / [Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2024-09-01**



Repository / Repozitorij:

[Algebra University - Repository of Algebra University](#)



VISOKO UČILIŠTE ALGEBRA

ZAVRŠNI RAD

**IMPLEMENTACIJA PROGRAMSKOG
RJEŠENJA ZA PODRŠKU UPRAVLJANJA
TENISKIM DOGAĐAJIMA**

Mijo Klepac

Zagreb, siječanj 2023.

„Pod punom odgovornošću pismeno potvrđujem da je ovo moj autorski rad čiji niti jedan dio nije nastao kopiranjem ili plagiranjem tuđeg sadržaja. Prilikom izrade rada koristio sam tuđe materijale navedene u popisu literature, ali nisam kopirao niti jedan njihov dio, osim citata za koje sam naveo autora i izvor, te ih jasno označio znakovima navodnika. U slučaju da se u bilo kojem trenutku dokaže suprotno, spreman sam snositi sve posljedice uključivo i poništenje javne isprave stečene dijelom i na temelju ovoga rada“.

U Zagrebu, 18.01.2023.

Predgovor

Zahvaljujem mentoru Danijelu Kučaku na prilici i strpljenju. Zahvaljujem mu na pomoći u odabiru i razradi teme te za pomoć pri izradi rada.

Sažetak

Danas internet služi kao osnovno sredstvo za elektroničku komunikaciju, razmjenu informacija, dodatno obrazovanje, pretragu informacija, za socijalnu komunikaciju, zabavu... Zbog užurbanog načina života i nedovoljno slobodnog vremena, aplikacija pomaže da se prati rezultati gdje god da se čovjek nalazi.

Ideja ovog rada je izrada web aplikacije koja će pomoći ljudima kako bi pratili svoje rezultate teniskih mečeva te spajanje ljubitelja navedenog sporta. Svrha izrade aplikacije je spojiti ljubitelje tenisa kako bi međusobno igrali, dogovarali susrete i pratili rezultate i svoj napredak.

Aplikacija je razvijena prema unaprijed predviđenim funkcionalnostima. Anketnim upitnikom, koji je proveden prije same izrade, utvrđeno je kako postoji interes za ovakvom vrstom aplikacije. Drugi anketni upitnik je proveden nakon napravljene aplikacije.

Praktični doprinos ovog rada je kompletno programsko rješenje za povezivanje teniskih igrača, organizaciju teniskih liga, te pregled povijesti odigranih susreta pojedinog igrača.

Ključne riječi: web aplikacija, tenis

Abstract

Today the internet is used as a basic tool for electronic communication, information exchange, additional education, search of information, for social networks, entertainment... Due to a busy lifestyle and insufficient free time, the application helps to track results wherever a person is.

The idea of this work was to create a web application that would help people track their tennis match results and connect fans of the said sport. The purpose of creating the application is to connect tennis enthusiasts to play with each other, arrange matches and track results and progress.

The application is developed according to predetermined functionalities. The survey, which was carried out before the creation itself, established that there was an interest in this type of application. The second survey was conducted after the application was made.

The practical contribution of this work will be a complete software solution for connecting tennis players, organizing tennis leagues, and reviewing the history of matches played by an individual player.

Keywords: Web application, tennis

Sadržaj

1. Uvod.....	1
2. Rekreativni sport.....	2
3. Situacijska analiza.....	6
3.1. Analiza konkurencije.....	7
3.2. Analiza potreba korisnika.....	8
4. Implementacija programskog rješenja.....	13
4.1. Dizajn podatkovnog sloja.....	14
4.2. Dizajn servisnog sloja.....	16
4.3. Dizajn prezentacijskog sloja.....	17
4.3.1. Korisničko sučelje.....	18
4.3.2. Prijava i sigurnost korisnika.....	19
4.3.3. Korisničko sučelje prijavljenog korisnika.....	19
4.3.4. Teniske lige.....	21
5. Korištene tehnologije.....	24
5.1. SQL.....	24
5.2. .NET Framework.....	24
5.3. C#.....	25
5.4. JavaScript.....	25
5.5. HTML.....	25
5.6. CSS.....	26
6. Analiza završne ankete ispitivanja zadovoljstva korisnika.....	27
Zaključak.....	30
Popis kratica.....	31
Popis slika.....	32
Popis tablica.....	33
Popis kôdova.....	34
Literatura.....	35

1. Uvod

Web aplikacije su programi koji su razvijeni u jezicima koje podržavaju suvremeni internet preglednici, kao što su Google Chrome, Microsoft Edge, Mozilla Firefox. Takve aplikacije se najčešće pišu u JavaScript i HTML jezicima. Na taj način Internet preglednik lako pročitava operacije koje su programirane. Web aplikacije su dostupne u bilo kojem trenutku, bilo na mobilnim telefonima ili na računalima. Kako bi aplikacija bila dostupna, obavezna je povezanost s internetom. Tako je i u ovoj aplikaciji, koja je dobila ime Tennis Vision, potreban internet kako bi bila dostupna.

Istraživanjem tržišta iz domene interesa, uočene su aplikacije kao što su Tennis Liga, Local Tennis Leagues i Oz Tennis League. Analizom navedenih aplikacija zaključeno je kako sve navedene aplikacije naplaćuju korisnicima korištenje te da ne postoji mogućnost vlastitog odabira igrača, kreiranje vlastite lige...

Kako bi se utvrdile potrebe tržišta, kreiran je i proveden anketni upitnik namijenjen osobama koje se bave tenisom, a ujedno imaju potrebu za praćenjem svojih rezultata te povezivanjem s drugim igračima.

Anketni upitnik je istraživački instrument koji se najčešće sastoji od niza pitanja u svrhu prikupljanja podataka od unaprijed definiranih ispitanika. U ovom radu primijenjen je anketni upitnik u dva oblika. Prvi anketni upitnik proveden je kako bi se vidjela potreba za izradom aplikacije, te su rezultati pokazali kako potreba postoji. Drugi anketni upitnik je proveden kako bi se ispitalo zadovoljstvo samom aplikacijom istih ispitanika.

Rad se bazira na:

1. Provedenom anketnom upitniku koji je proveden prije početka pisanja rada i izrade aplikacije
2. Implementiranom programskom rješenju
3. Provedenom i analiziranom završnom anketnom upitniku zadovoljstva korisnika implementiranog programskog rješenja.

2. Rekreativni sport

Rekreativni sport obuhvaća vrste aktivnosti koje se događaju u slobodno vrijeme čovjeka. Te aktivnosti doprinose razvoju stvaralačkog potencijala, zdravlja, životnog optimizma, zadovoljstva i kvalitete života čovjeka [1].

Ne računajući kisik, vodu i energiju, mišićna aktivnost i kretanje su jedne od najosnovnijih ljudskih potreba. Spoznaje i istraživanja pokazuju kako kvalitetan način življenja podrazumijeva pored nepušenja, ovisnosti o alkoholu i drogama, zdravog načina prehrane, otpornosti prema stresu i infekcijama i tjelesno vježbanje kao važan dio u stvaranju zdravog ljudskog organizma [2].

Današnji način života u kojem je život bez tehnologije nezamisliv, doveo je do toga da čovjekova aktivnost tijekom dana nije dovoljna. Na svjetskoj razini sport i sportska rekreacija označeni su kao jedan od najvažnijih područja društvenog razvoja te mnoge zemlje sportsku rekreaciju smatraju bitnim čimbenikom određenja kvalitete života, zdravlja i radne učinkovitosti građana [3].

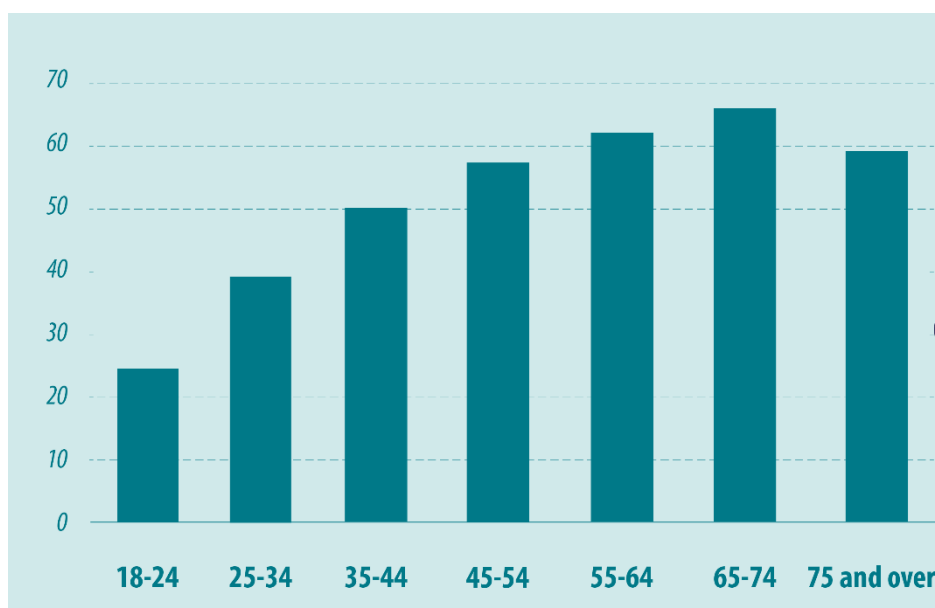
Preporučena tjedna tjelesna aktivnost u odraslih je barem 150 minuta tjelesne aktivnosti umjerenog intenziteta ili 75 minuta tjelesne aktivnosti visokog intenziteta [4]. Umjerena tjelovježba povećava energetske potrebe, povećava količinu HDL kolesterola (dobar za zdravlje), smanjuje vrijednost LDL kolesterola (loš za zdravlje), snižava krvni tlak, povećava osjetljivost na inzulin s manjim stupnjem kardiovaskularnih oboljenja, dijabetesa tip 2, osteoporoze, tumora debelog crijeva i dojke...

Pod rekreacijske sportove u pravilu pripadaju sve one aktivnosti kod kojih je primarna svrha aktivnosti sudjelovanje, a ne natjecanje. Ostali ciljevi rekreacijskog sporta su poboljšanje tjelesne aktivnosti i društvena uključenost. Rekreativni sportovi, za razliku od natjecateljskih, donose i puno manje stresa za sudionike sporta te nemaju rigorozne treninge.

Također, većina zdravstvenih agencija preporuča bavljenje sportom zbog boljeg psihičkog i fizičkog zdravlja. Rekreacijski sport, kako kod mladih tako i kod odraslih osoba, napreduje kao sastavnica zdravog načina života koji dovodi do javljanja manjeg broja obolijevanja od teških bolesti (osobito dijabetesa i kardiovaskularnih bolesti povezanih s pretilošću) te općenito veće dugovječnosti stanovništva.

Današnji užurbani stil života nam onemogućuje dovoljno bavljenja sportom te se iz sljedećih grafikona vidi koliki postotak ljudi ima prekomjernu tjelesnu težinu.

Iz *Slika 2.1* je vidljivo da što je dobna skupina starija, to je veći udio osoba koje imaju prekomjernu tjelesnu težinu. Najmanji postotak osoba s prekomjernom težinom zabilježen je kod osoba koje se nalaze između 18-te i 24-te godine života (25%), a najveći postotak je zabilježen kod osoba od 65-te do 74-te godine (66%) [5]. Skoro polovica odraslih koji žive u EU (45%) je imalo normalnu tjelesnu težinu 2019. godine, a malo više od polovice (53%) smatralo se da imaju prekomjernu tjelesnu težinu. Rezultati su dobiveni prema indeksu tjelesne mase [5]. Indeks tjelesne mase (BMI) je metoda izračuna uhranjenosti. BMI se temelji na odnosu težine i visine osobe.

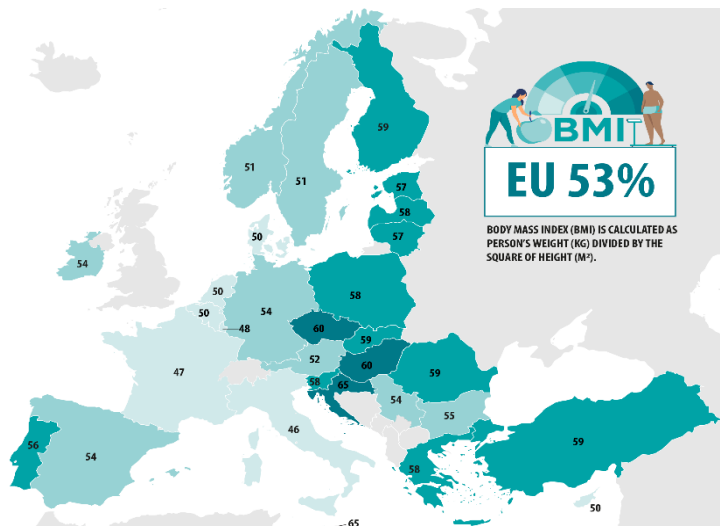


Izvor: <https://ec.europa.eu/eurostat/web/products-eurostat-news/-/ddn-20210721-2>

Slika 2.1. Postotak stanovništva s prekomjernom tjelesnom težinom u EU, prema dobi

Udio odraslih osoba s prekomjernom tjelesnom težinom razlikuje se među državama članicama EU, a najveći udio zabilježen je u Hrvatskoj i na Malti, gdje se u 2019. godini 65% odraslih osoba smatralo pretilima.

Prema dostupnim podacima, vidljivo je da se Hrvatska nalazi na samom vrhu po pretilosti. U tom smislu, rekreacijski sport je vrlo važan, jer može pospješiti smanjenje udjela pretilih osoba i općenito doprinijeti zdravijem načinu života ljudi.



Izvor: <https://ec.europa.eu/eurostat/web/products-eurostat-news/-/ddn-20210721-2>

Slika 2.2. Postotak stanovništva s prekomjernom tjelesnom težinom po državama

Tenis je jedan od najpopularnijih individualnih sportova u svijetu [6]. U razvijenim društvima osnova za stjecanje društvenog statusa vidi se u obrazovanju, međutim, u istočnim i jugoistočnim europskim zemljama mnoge obitelji ne vide stjecanje društvenog statusa kroz obrazovanje, nego kroz sport. Čest odabir sportskih aktivnosti takvih obitelji je upravo tenis, jer se tenis smatra sportom za „elitu“. S druge strane, u tenisu je prijelaz iz amaterskih u profesionalna natjecanja težak i dugotrajan proces. Rezultati pokazuju da 91% među 20 najbolje rangiranih dječaka postiže profesionalne rezultate [6].

Tenis je igra u kojoj ili dva pojedinca ili četiri člana tima (parovi) udaraju reketom lopticu preko mreže koja se proteže cijelom širinom terena. Igra se promijenila od vremena kad je nastala do danas. Tenis su u srednjem vijeku igrali francuski redovnici i plemstvo. Moderan tenis je nastao u Engleskoj kasnih 1880-ih [7]. Francuski redovnici su dlanovima udarali lopte po mreži u dvorani, dok je danas tenis međunarodni sport u kojem uživaju deseci milijuna ljudi diljem svijeta.

Moderan tenis, otprilike kakav danas poznajemo, izumljen je kada je Velšanin Walter Clopton Wingfield prilagodio „stari“ tenis. Kako se tenis u Francuskoj igrao u zatvorenom prostoru, Walter Clapton Wingfield ga je „preselio“ na otvoreni prostor (danas se to zove tenis na travnjaku) [7].

Pravila igre su takva da tenis igraju dva igrača (ako je pojedinačna igra), odnosno četiri igrača (ako je igra parova). Osnovni je cilj igre reketom uputiti lopticu preko mreže na stranu protivnika na takav način da ju protivnik ne uspije vratiti prije nego što loptica drugi put dodirne površinu. Igra počinje servisom, kojeg prvi izvodi igrač određen ždrijebom ili bacanjem

novčića. Servis se izvodi iza zadnje linije vlastitog polja, a lopticom se mora pogoditi dijagonalno servisno polje na protivničkoj strani terena. Poen se osvaja ukoliko protivnik ne uspije vratiti lopticu koja je pala u njegov dio terena preko mreže ili ako protivnik izbacilopticu izvan terena prije nego što prvi puta dodirne površinu [8].

3. Situacijska analiza

Situacijska analiza odnosi se na skup metoda koje menadžeri koriste za analizu unutarnjeg i vanjskog okruženja organizacije kako bi razumjeli sposobnosti, klijente i poslovno okruženje tvrtke [9]. Svrha takve analize je otkrivanje prednosti i slabosti neke tvrtke, a analiza makro i mikro okolišnih čimbenika često može ukazati koje se strategije moraju slijediti. Analiza situacije je ključan korak za uspostavljanje dugoročnog odnosa s klijentima. U korporativnom životu situacijska analiza pomaže u definiranju trenutne situacije i djelovanju kako bi se nastavilo napredovati [10]. Današnji način života i brz razvoj tehnologije zahtijevaju analizu koja može dati sliku kako tvrtka stoji u poslovnom okruženju, također može i poboljšati rast tvrtke te predstaviti neke nove razvojne mogućnosti.

SWOT analiza (engl. *Strengths, Weaknesses, Opportunities, Threats*) je okvir koji se koristi za ocjenu konkurentske pozicije poduzeća i razvoj strateškog planiranja. SWOT analiza procjenjuje unutarnje i vanjske čimbenike, kao i trenutni i budući potencijal. Osmišljena je kako bi omogućila realan, činjenicama temeljen pogled na snage i slabosti organizacije, prilike i prijetnje unutar njezine industrije. Organizacija mora održavati analizu točnom, izbjegavajući unaprijed zamišljena uvjerenja ili sive zone i umjesto toga fokusirajući se na kontekste stvarnog života [11].

Za ovaj projekt je korištena SWOT analiza kako bi se otkrile snage, slabosti, prilike i prijetnje projekta.

Na *Slika 3.1* je prikazano kako je osoba zadužena za razvoj aplikacije stekla dovoljno znanja tijekom studija za razvoj ovakve aplikacije. Također, istraživanjem tržišta je uočeno kako broj ovakvih aplikacija nije velik. Korisnici aplikacije su osobe koje se bave tenisom, stoga je aplikacija korisna za određen broj ljudi. Međutim, postoji mogućnost širenja aplikacije na ostale sportove. Na taj način aplikacija bi postala korisna za veći broj korisnika. Velika prednost aplikacije je i mogućnost igranja s ostalim korisnicima aplikacije, što podrazumijeva i upoznavanje drugih igrača. Aplikacija je nova, te se treba razvijati i u budućnosti.



Slika 3.1. SWOT analiza

3.1. Analiza konkurencije

Istraživanjem i analizom tržišta, pronađene su slične aplikacije. Neke od njih su:

- **Tenis Liga** – ovoj ligi se mogu pridružiti svi koji imaju stečeno znanje u tenisu. Igrači se razvrstavaju u lige po znanju igranja. Aplikacijom upravlja „upravni odbor“ te određuje koji igrač ide u koju ligu [12].
- **Local Tennis Leagues** – ova aplikacija je namijenjena stanovnicima Velike Britanije. Za navedenu aplikaciju je potrebno prethodno znanje. Svaka osoba napravi profil i dobije LTA broj (Local Tennis Leagues broj). Spajanje igrača određuju kreatori aplikacije [13].
- **Oz Tennis League** – ova aplikacija ima online obrazac. To je obrazac u kojem treba popuniti informacije o svom igračkom iskustvu kako bi se mogli spojiti s igračima koji imaju otprilike jednako znanje u tenisu. Spajanje parova određuju kreatori aplikacije [14]. I ova aplikacija također traži prethodno znanje u tenisu.

Niti jedna od navedenih aplikacija nema mogućnost vlastitog odabira igrača, svaka aplikacija se naplaćuje, te ne postoji mogućnost kreiranja vlastite lige što je vidljivo u *Tablica 3.1.*

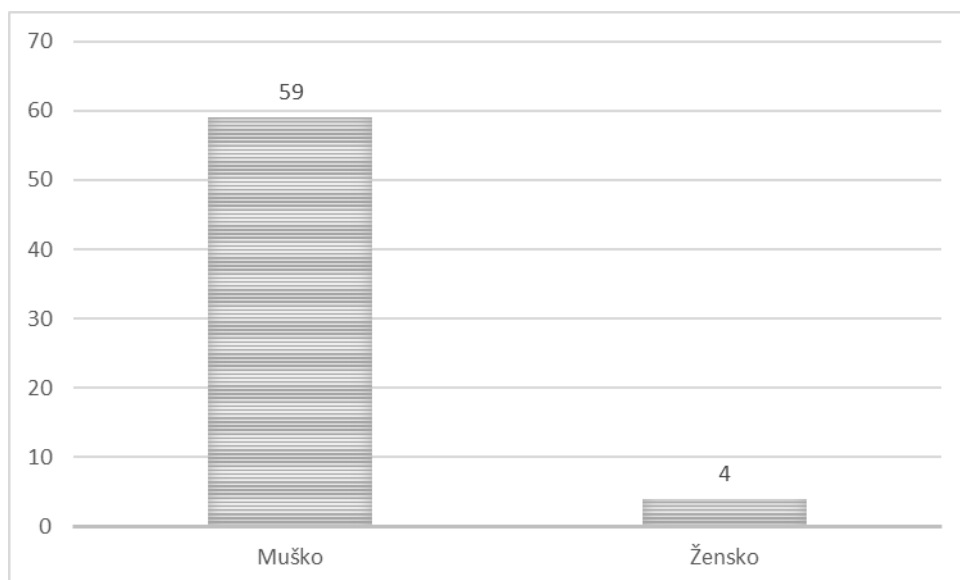
	Tennis Vision	Tenis Liga	Local Tennis Leagues	Oz Tennis League
Potrebno znanje u tenisu	Ne	Da	Da	Da
Plaćanje aplikacije	Ne	Da	Da	Da
Vlastiti odabir igrača	Da	Ne	Ne	Ne
Kreiranje vlastite lige	Da	Ne	Ne	Ne
Pretraživanje drugih igrača	Da	Ne	Ne	Ne

Tablica 3.1. Usporedba konkurencije

3.2. Analiza potreba korisnika

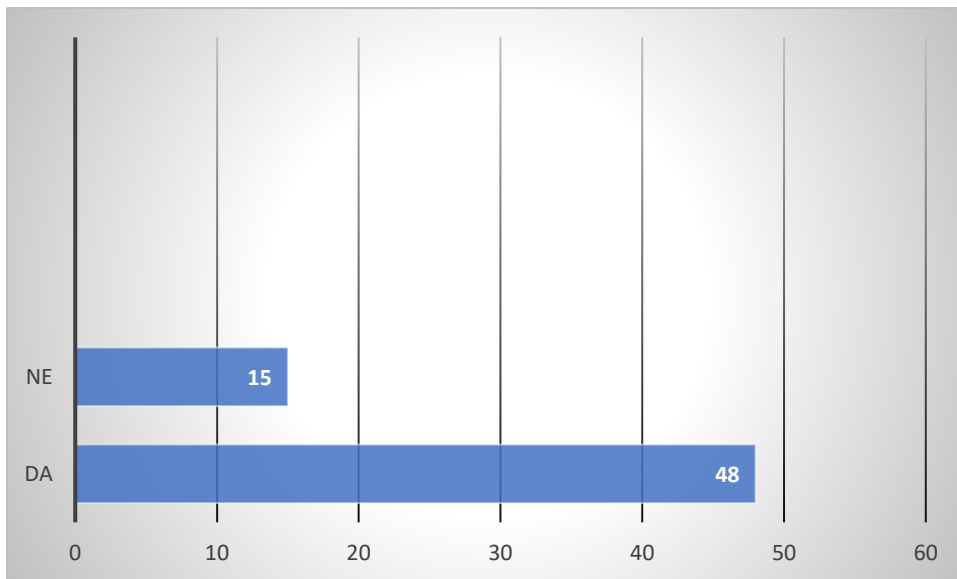
Provedeni anketni upitnik je pomogao prikupiti informacije za potrebom izrade ovakve vrste aplikacije. Ukupno 63 ispitanika je ispunilo anketni upitnik. Anketni upitnici su podijeljeni prilikom posjeta teniskim centrima u Zagrebu.

Slika 3.2 prikazuje broj ispitanika prema spolu. Vidljivo je kako su većina ispitanika, njih 59, odnosno 94%, muškarci, dok su samo 4 (6%) ispitanika ženskog spola. Po navedenom se može zaključiti kako se više osoba muškog spola rekreativno bavi tenisom, nego osobe ženskog spola.



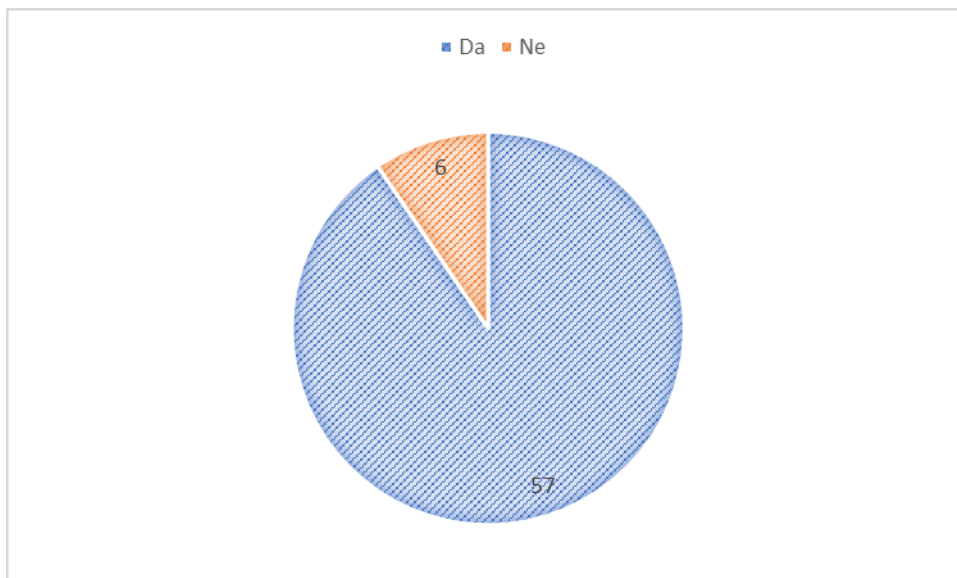
Slika 3.2. Spol ispitanika

Najveći postotak ispitanika (60%) zaokružilo je da se nalazi između 25 i 34 godine života. Iz Slika 3.3 je vidljivo kako je najviše ispitanika mlađe životne dobi, što pokazuje da se još uvijek i mladi bave rekreacijom.



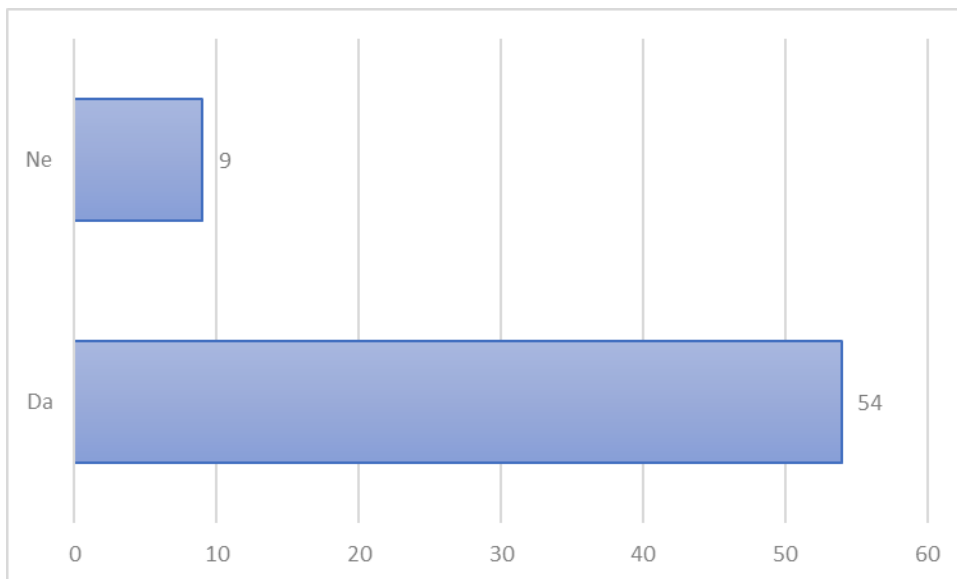
Slika 3.3. Dob ispitanika

Prema prikazanoj slici (*Slika 3.4*) vidljivo je da je 57 ispitanika, odnosno 90% na anketnom upitniku zaokružilo Da na pitanje „Igrate li tenis rekreativno?“. Puno manji broj ispitanika, samo njih dvoje, zaokružilo je odgovor Ne. Današnji užurbani način života nam rijetko daje mogućnost viška slobodnog vremena, no ipak se pokazuje da ljudi uspijevaju pronaći malo vremena za rekreaciju.



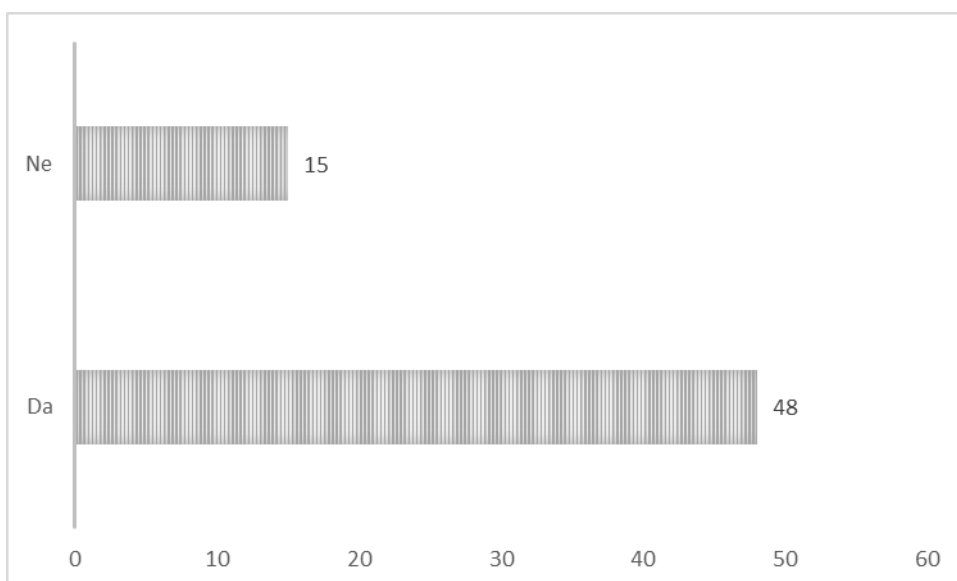
Slika 3.4. Igrate li tenis rekreativno?

Na pitanje „Sudjelujete li u ikakvim amaterskim natjecanjima?“, 54 ispitanika (86%) odgovorilo je kako sudjeluju. Ostalih devet ispitanika (14%) su zaokružila odgovor Ne. Ciljana skupina su bile osobe koje sudjeluju u takvoj vrsti natjecanja kako bi se mogla ispitati i potreba za ovom aplikacijom.



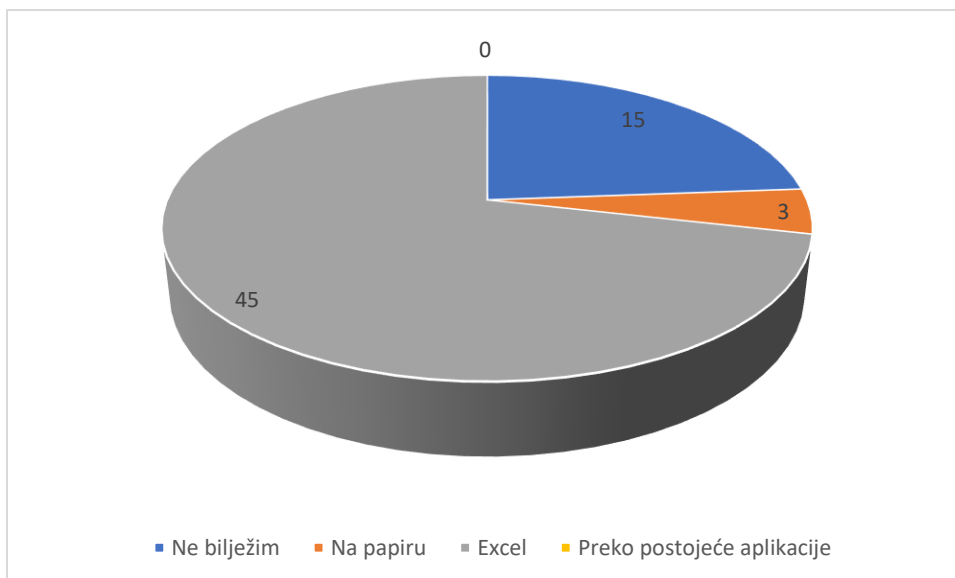
Slika 3.5. Sudjelujete li u ikakvim amaterskim natjecanjima?

Sljedeće pitanje je bilo postavljeno na način da se ispita bilježenje odigranih mečeva kako bi se vidjelo ima li smisla raditi aplikaciju ukoliko ispitanici ne bilježe svoje rezultate. Od 63 ispitanika, 48, odnosno 76% ih je odgovorilo potvrdno, dok je 24%, odnosno 15 ispitanika odgovorilo kako ne bilježi svoje rezultate.



Slika 3.6. Bilježite li rezultate odigranih mečeva u bilo kojem obliku?

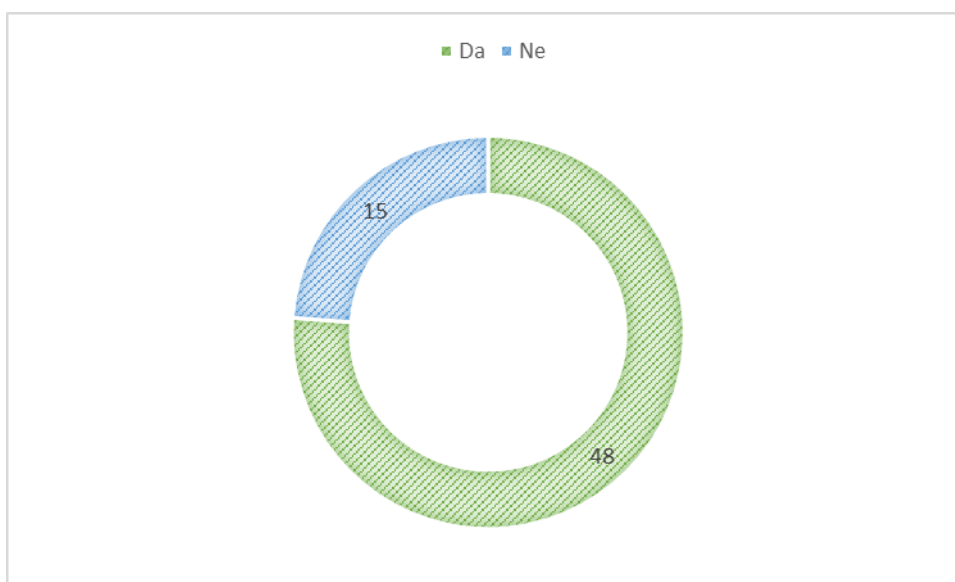
Slika 3.7 prikazuje odgovor na pitanje „Na koji način bilježite rezultate odigranih mečeva?“. Većina, čak 45 ispitanika, odnosno 71%, je na ovo pitanje odgovorilo kako bilježe rezultate pomoću programa Excel. Nekolicina, njih 15 (24%), odgovorilo je da svoje rezultate ne bilježe nigdje. Tri ispitanika (5%) su odgovorila kako rezultate bilježe na papiru, dok niti jedan ispitanik (0%) ne bilježi rezultate pomoću aplikacije predviđene za to.



Slika 3.7. Bilježite li rezultate odigranih mečeva u bilo kojem obliku?

Sljedeće pitanje u anketnom upitniku je bilo da ispitanici navedu ime aplikacije u kojoj bilježe svoje rezultate, međutim, niti jedan ispitanik ne bilježi rezultate preko aplikacije, tako da na ovo pitanje nije bilo odgovora.

Posljednje pitanje koje je prikazano na *Slika 3.8* je bilo „Bi li Vas zanimala aplikacija za organizaciju vlastite lige te praćenje rezultata svojih odigranih mečeva?“. Na ovo pitanje je 48 ispitanika, odnosno 76%, odgovorilo kako bi ih zanimala ovakva aplikacija. Petnaest ispitanika je odgovorilo kako ih ne bi zanimala takva aplikacija.



Slika 3.8. Bi li Vas zanimala aplikacija za organizaciju vlastite lige te praćenje rezultata svojih odigranih mečeva?

Iz provedenog anketnog upitnika možemo zaključiti kako niti jedan ispitanik ne koristi aplikaciju koja bi bilježila rezultate mečeva. Većina ispitanika bilježi svoje rezultate pomoću improvizirane Excel tablice. Također, zaključuje se kako bi većina ispitanika bila zainteresirana za takvu aplikaciju.

4. Implementacija programskog rješenja

Postoji više vrsta arhitektura aplikacije a najčešće se koristi tzv. troslojna arhitektura. Troslojna arhitektura je arhitektura tipa klijent-poslužitelj u kojoj se logika funkcionalnog procesa, pristup podacima, pohrana računalnih podataka i korisničko sučelje razvijaju i održavaju kao neovisni moduli na zasebnim platformama [15].

Postoje tri razine u troslojnoj arhitekturi [15]:

- **Prezentacijski sloj**
 - Zauzima najvišu razinu i prikazuje informacije povezane s uslugama koje su uobičajeno dostupne u web pregledniku ili web aplikaciji u obliku grafičkog korisničkog sučelja (GUI).
 - Obično je izgrađen na okvirima za web razvoj, kao što su CSS ili JavaScript
 - Komunicira s drugim razinama slanjem rezultata pregledniku i drugim razinama u mreži putem API poziva.
- **Servisni sloj**
 - Kontrolira temeljnu funkcionalnost aplikacije izvodeći detaljnu obradu i obično je kodiran u programskim jezicima, kao što su Python, Java, C++, .NET itd.
- **Podatkovni sloj**
 - Podaci se čuvaju neovisno o aplikacijskim poslužiteljima ili poslovnoj logici, a njima se upravlja i njima se pristupa pomoću baza podataka kao što su MongoDB, Oracle, MySQL i Microsoft SQL Server.



Slika 4.1. Troslojna arhitektura aplikacije

4.1. Dizajn podatkovnog sloja

Podatkovni sloj aplikacije zadužen je za spremanje svih podataka bitnih za aplikaciju te ujedno i za njihovo dohvaćanje. U ovoj aplikaciji korišten je Microsoft SQL Server za potrebe spremanja podataka.

Baza Podataka

Microsoft SQL Server je relacijski tip baze podataka što znači da su međusobno ovisne tablice povezane relacijama kako bi se osigurala točnost i sigurnost pohranjenih podataka. Kao što i samo ime kaže Microsoft SQL Server je izgrađen na bazi SQL-a. SQL je je programski jezik koji se koristi za upravljanje bazama podataka te za izvršavanje upita nad bazom.

Za potrebe ovoga rada, izrađen je ER dijagram koji grafički prikazuje odnose između tablica te samu strukturu baze podataka ove aplikacije. Baza podataka sastoji se od 16 tablica, od kojih svaka sadrži svoj primarni ključ koji mora biti jedinstven za svaki red u tablici te se ne smije ponavljati. Nazivi tablica su osmišljeni tako da direktno opisuju koji podaci se nalaze u tablici.

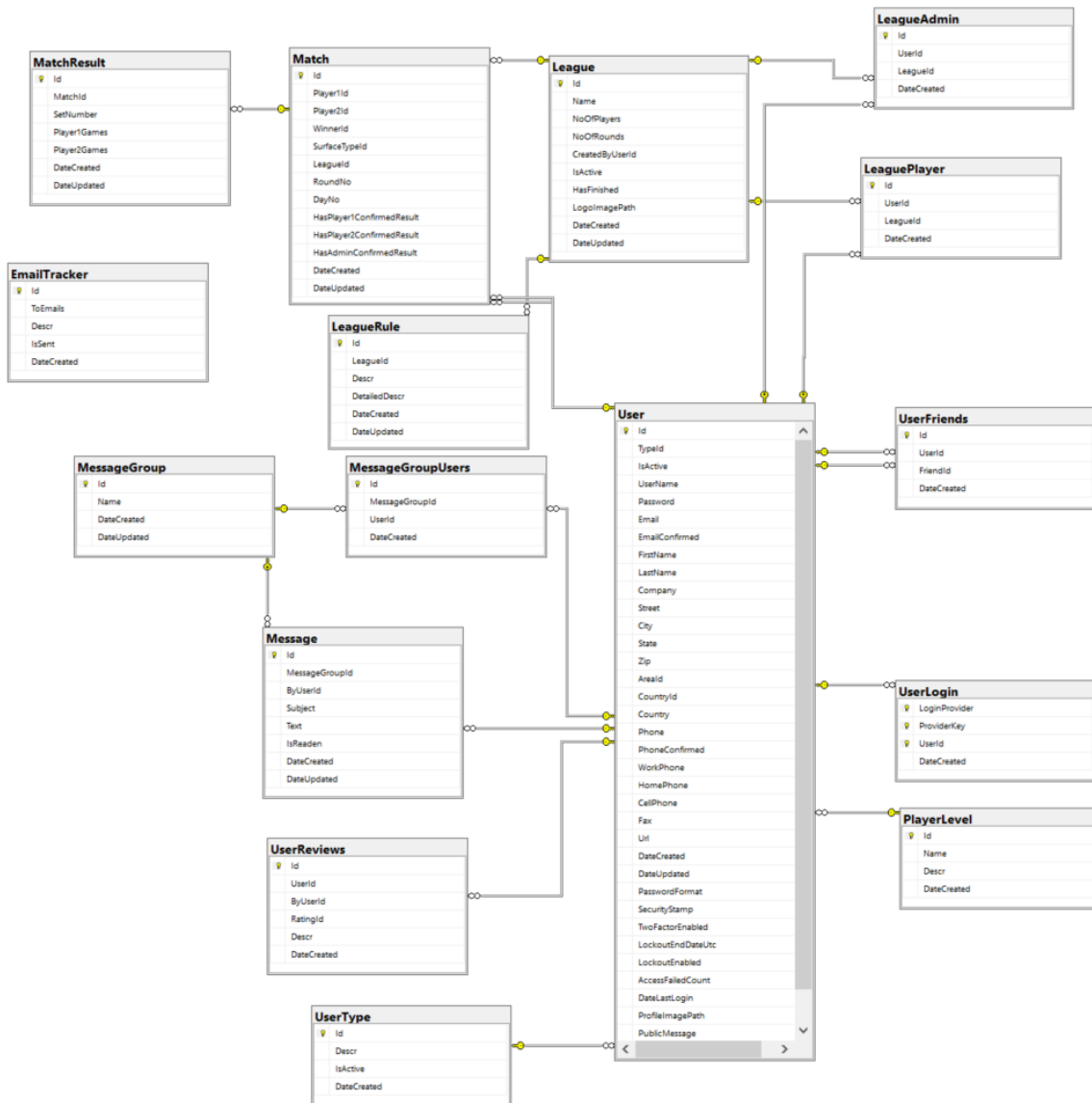
NAZIV TABLICE	KRATKI OPIS
EmailTracker	Tablica koja je zamišljena da samo prati broj dnevno poslanih e-mailova
League	Sadrži osnovne podatke o kreiranoj teniskoj ligi kao što su ime, broj igrača, broj kola koji će se odigrati
LeagueAdmin	Tablica u kojoj se nalazi tko je administrator za pojedinu ligu
LeaguePlayer	Tablica u koju se spremaju podaci o tome koji igrač sve sudjeluje u pojedinoj teniskoj ligi
LeagueRule	Korisnici imaju mogućnost prilikom kreiranja lige upisati neka pravila koja su specifična za njihovu ligu i ti podaci se spremaju u ovu tablicu
Match	Prilikom kreiranja lige, te nakon što se odrede igrači koji sudjeluju u ligi kreiraju se svi mečevi koji trebaju biti odigrani, a ti podaci se zapisuju u ovu tablicu
MatchResult	Ova tablica služi za spremanje rezultata pojedinog meča. Mečevi mogu biti odigrani od 1 do 5 setova, a jedan zapis u ovoj tablici je zapravo 1 odigrani set
Message	Tablica koja služi za spremanje poruka koje korisnici mogu razmjenjivati
MessageGroup	Kada korisnik želi poslati nekome poruku kreira se grupa, a onda korisnici koji su dodani u tu grupu imaju mogućnost razmjenjivati poruke
MessageGroupUsers	Tablica koja služi za spremanje podataka koji se sve korisnici nalaze u pojedinoj grupi
PlayerLevel	Ovo je statička tablica gdje su spremljeni podaci o tome koje su dostupne razine igranja tenisa. To znači da korisnik sam za sebe određuje koliko je dobar igrač, a taj podatak se zapisuje u tablicu User
User	Tablica u kojoj se nalaze svi osnovni podaci o korisnicima, kao što su e-mail, lozinka, tip, ime, prezime
UserFriends	Korisnik može dodavati prijatelje, a ti podaci se zapisuju u ovu tablicu
UserLogin	Podaci o prijavi korisnika u aplikaciju koji se spremaju u ovu tablicu
UserReviews	Korisnici mogu ostavljati recenzije o drugim igračima te ostaviti ocjenu što oni smatraju koja je razina igranja tenisa tog igrača
UserType	Tablica u kojoj su zapisani podaci koji sve tipovi korisnika postoje u aplikaciji

Tablica 4.1 Tablice u bazi podataka

Postoje tri glavne tablice. Prva je tablica „User“. Tu su spremljeni glavni podaci o korisnicima koji se registriraju u aplikaciji. Podaci potrebni za kasniji ulazak u aplikaciju kao što su korisničko ime i lozinka, također su spremljeni u ovoj tablici, s time da je bitno napomenuti kako su lozinke kriptirane da se takvi podaci ne bi mogli koristiti na neželjeni način.

Druga bitna tablica za aplikaciju je „League“. To je glavna tablica u koju se bilježe podaci o kreiranim ligama, na koju se veže još par tablica ključnih za teniske lige.

Treća tablica je tablica „Match“. Tu se nalaze svi mečevi za pojedinu ligu te se na nju veže još jedna bitna tablica „MatchResult“. Ta tablica je jako bitna jer aplikacija omogućuje korisnicima da sami evidentiraju koliko su setova u teniskom meču odigrali te da upišu odigrani rezultat meča.



Slika 4.2. Dijagram baze podataka

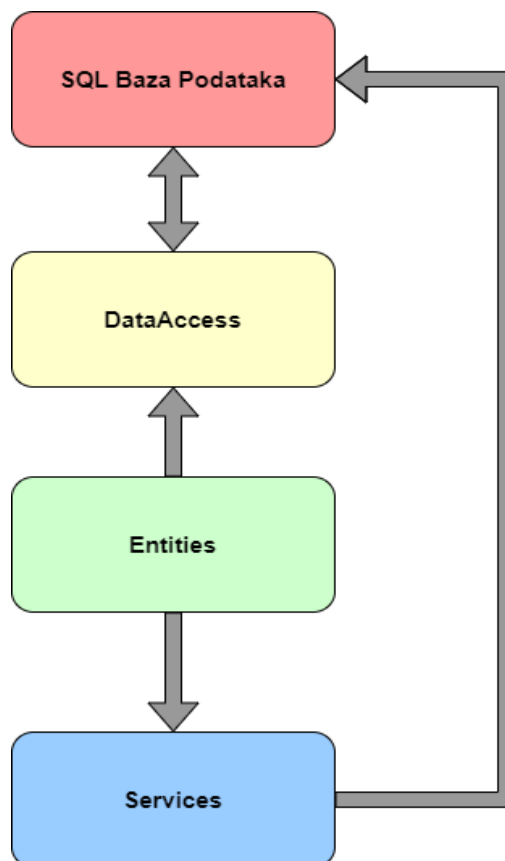
4.2. Dizajn servisnog sloja

Servisni sloj aplikacije napisan je u programskom jeziku C# te se sastoji od tri različita projekta tipa „ClassLibrary“. Nazivi tih projekata su „DataAccess“, „Entities“ i „Services“.

„DataAccess“ je projekt napravljen iz razloga da njegova glavna zadaća bude komunikacija s bazom podataka. Za kreiranje dinamičkih upita prema SQL bazi korišten je programski jezik C#.

Projekt „Entities“ nema nikakvu posebnu specifičnu logiku nego se samo sastoji od klasa koje su preslike tablica iz baze podataka. Ovaj projekt je bio potreban da svaka promjena u kôdu bude što jednostavnija, ukoliko dođe do nekih bitnih promjena na dizajnu tablica u bazi podataka kako bi svi ostali slojevi mogli koristiti iste klase.

Projekt „Services“ je napravljen kao poseban sloj koji služi za komunikaciju između klijentskog dijela aplikacije i sloja kojemu je glavna zadaća komunikacija s bazom podataka. Razlog zbog kojeg je došlo do toga je ukazivanje potrebe za dodatnim pisanjem kôda prije nego što su određeni podaci bili spremni da budu upisani u bazu podataka. Ta potreba je bila najizraženija u trenutku kada se kreiraju mečevi za pojedinu tenisku ligu. Tu je trebalo razraditi posebnu logiku iz razloga što u ligi postoji mogućnost da bude neparan broj igrača, različit broj kola, a taj se dio razrađuje upravo u projektu „Services“. Samim time taj dio nije mogao biti napisan niti u klijentskom dijelu aplikacije, a niti u „DataAccess“ sloju koji služi samo za komunikaciju s bazom podataka.

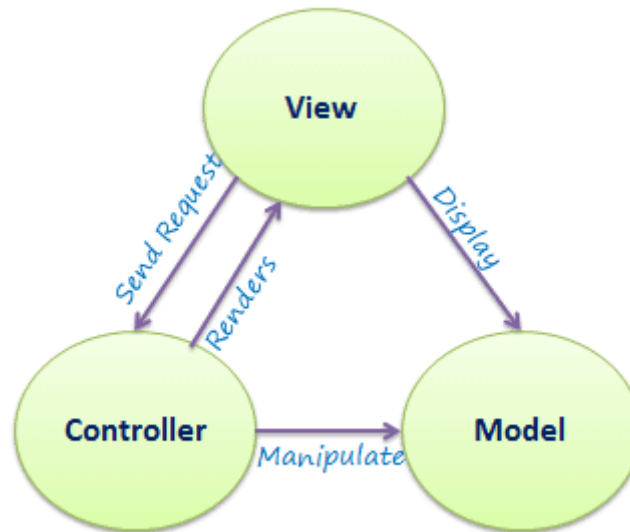


Slika 4.3. Servisni sloj

4.3. Dizajn prezentacijskog sloja

Prezentacijski sloj implementiran je u MVC .Net Frameworku koristeći HTML, CSS, JavaScript i C#. On je web radni okvir temeljen na arhitekturi Model-View-Controller (MVC).

MVC dijeli aplikaciju u tri glavne skupine. To su modeli, kontroleri i stranice. Kada se koristi MVC obrazac klijent šalje zahtjev prema serveru i do kontrolera koji želi. Nakon toga kontroler klijentu vraća stranicu koja sadrži željene podatke. Ovakav način programiranja, gdje se lakše podijele zadaće pomaže u brzem ispravljanju pogrešaka te u kasnijem modificiranju kôda.



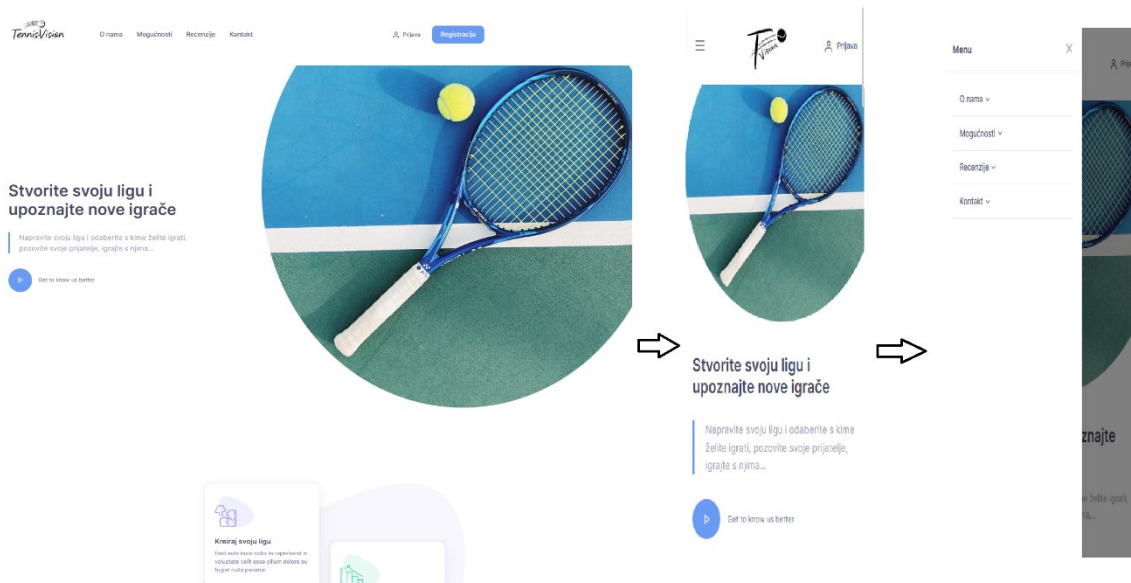
Izvor: <https://www.tutorialsteacher.com/mvc/mvc-architecture>

Slika 4.4. MVC arhitektura

4.3.1. Korisničko sučelje

Za izradu vizualnog dijela aplikacije korištene su tehnologije HTML, CSS i JavaScript. U izradi dizajna pazilo se da izgled stranice bude responzivan a to znači da korisnik može imati dobro iskustvo u aplikaciji na svim veličinama ekrana. Kako bi se to postiglo korišten je Bootstrap verzija 5. Bootstrap je omogućio već gotove CSS klase koje su ubrzale izradu ove aplikacije, ali i pomogle da korisnik može imati najbolje moguće iskustvo prilikom korištenja aplikacije.

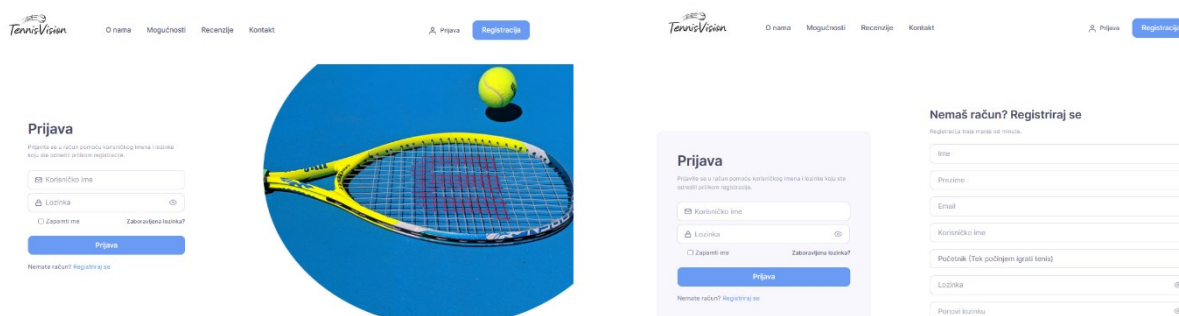
Bootstrap je besplatni front-end razvojni okvir otvorenog kôda za izradu web stranica i web aplikacija [17]. Bootstrap uključuje osnove responzivnog web razvoja, tako da programeri trebaju umetnuti unaprijed kreirane klase kako bi HTML elementi poprimili ta oblikovanja.



Slika 4.5. Početni ekran Tennis Vision

4.3.2. Prijava i sigurnost korisnika

Za potrebe prijave i registraciju korisnika u aplikaciju korišten je ASP .NET Identity. On omogućuje dodavanje prilagođene funkcionalnosti prijave/odjave i prilagođenih značajki profila koje olakšavaju prilagodbu podataka o prijavljenom korisniku. Autentifikacijom korisnik potvrđuje kako je to stvarno on, a autorizacijom server omogućuje pristup na temelju postojanja privilegija ukoliko je autentifikacija uspješno odrađena. Isto tako ASP .NET Identity već ima gotove funkcionalnosti koje štite korisnika kao što je spremanje lozinki u bazu podataka. Jako je bitno da nitko ne može vidjeti lozinku u originalnom obliku pa se lozinka u bazu pohranjuje kriptirana. To daje sigurnost korisniku da nitko drugi ne može vidjeti njegovu izvornu lozinku.



Slika 4.6. Prijava/registracija

4.3.3. Korisničko sučelje prijavljenog korisnika

Jednom kada se korisnik prijavi u sustav, on koristi drugačiji izgled stranice nego što je prikazano na početnoj stranici. Za tu namjenu napravljen je potpuno novi izgled naziva

„_UserLayout“. Sve stranice u aplikaciji su rađene na način da se kombinirao HTML i Razor mogućnosti koje nudi MVC .Net Framework. Korisnik se nakon uspješno obavljene prijave u aplikaciju preusmjerava na stranicu na kojoj je omogućena promjena vlastitih podataka, te dodavanje ili promjena profilne fotografije. Važno je napomenuti kako se svaka učitana slika u aplikaciju ujedno i komprimira. To je napravljeno iz razloga što današnje slike zbog svoje visoke rezolucije zauzimaju puno prostora. To je predstavljalo problem kad bi se te slike htjele prikazati korisniku jer bi učitavanje stranica trajalo dugo. Pomoću već gotovog „NuGet“ paketa „magick.NET.Core“ napravljeno je smanjivanje slika. U svrhu toga napravljena je i klasa u aplikaciji koja se zove „ImageHelper“ i njena metoda „ResizeWithMagickImage“. Dodavanje te funkcionalnosti je znatno ubrzalo učitavanje stranica dok se kvaliteta učitanih slika nije puno promijenila. Primjer kôda te funkcionalnosti:

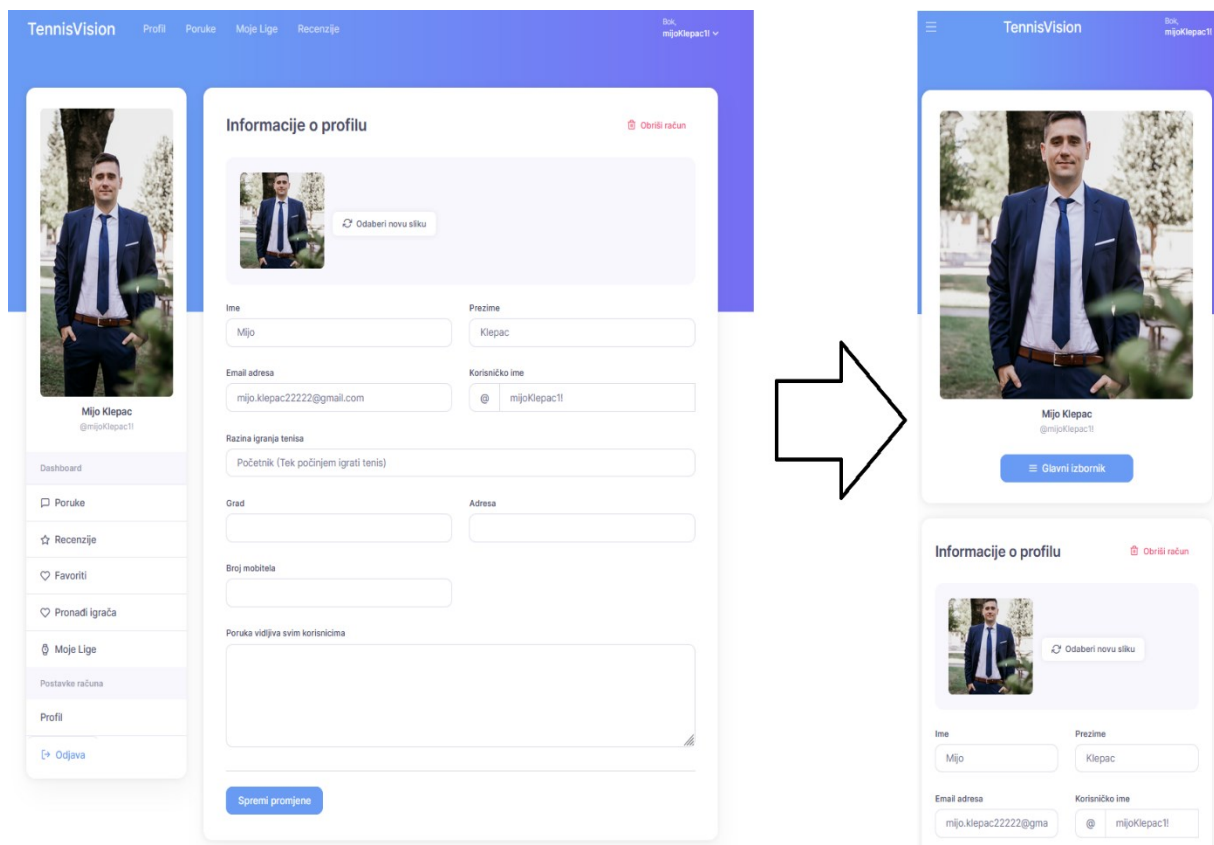
```
public static Image ResizeWithMagickImage(Stream inputStream)
{
    using (MagickImage i = new MagickImage(inputStream))
    {
        i.Format = i.Format; // Get or Set the format of the image.

        var imageSize = (i.ToArray().Length / 1024) / 1024.0;
//size in MB
        if (imageSize > 10)
            i.Quality = 70; // This is the Compression level.
        else if (imageSize > 8)
            i.Quality = 75; // This is the Compression level.
        else if (imageSize > 6)
            i.Quality = 85; // This is the Compression level.
        else if (imageSize > 4)
            i.Quality = 90; // This is the Compression level.
        else if (imageSize > 2)
            i.Quality = 95; // This is the Compression level.
        else
            i.Quality = 100; // This is the Compression level.

        return (Bitmap)((new
ImageConverter()).ConvertFrom(i.ToArray()));
    }
}
```

Kôd 4.1. Kôd za smanjivanje slika

Izgled na stranicama je responzivan kako bi se omogućilo dobro iskustvo na svim veličinama ekrana.



Slika 4.7. Prikaz stranice nakon uspješne prijave u aplikaciju

4.3.4. Teniske lige

Glavne funkcionalnosti ove aplikacije su kreiranje teniskih liga, unošenje rezultata mečeva te praćenje tih istih rezultata. Svaki registrirani korisnik ima mogućnost kreiranja svoje teniske lige. To može napraviti na jednostavan način tako da u izborniku nakon klika na „Moje Lige“ izabere opciju kreiranje nove teniske lige ili da pregleda neku već postojeću. Prilikom kreiranja teniske lige korisnik prvo mora odrediti naziv te broj kola koliko se planira odigrati unutar lige. Minimalan broj kola je 1, a maksimalan 15. Broj kola određuje koliko će mečeva međusobno odigrati igrači. Nakon toga potrebno je učitati i sliku koja će služiti kao logo teniske lige. Tada dolazi do dodavanja sudionika u ligu. To se izvršava na način da kreator upisuje korisničko ime a pri tome mu pomaže automatsko dovršavanje. Za tu namjenu isprogramirana je funkcionalnost automatskog dovršavanja gdje korisnik prilikom upisivanja imena igrača može odmah vidjeti postoji li uopće takav igrač kao registrirani korisnik. U teniskim ligama se ne može sudjelovati ukoliko niste registrirani korisnik u aplikaciji. Ukoliko je prilikom dodavanja igrača kreator napravio pogrešku ima mogućnost odmah maknuti tog igrača iz lige. Kreator lige se automatski smatra sudionikom lige te je obavezno nadodati barem još jednog igrača. Nakon odabira igrača aplikacija ima algoritam koji kreira mečeve za tu ligu tako da igrači međusobno igraju onoliko puta koliko su odredili na početku (broj kola). Algoritam je napisan u servisnom

sloju aplikacije. Svi kreirani mečevi se zapisuju odmah u bazu podataka kako bi korisnici mogli u svakom trenutku pratiti tko bi s kime trebao igrati. Primjer kôda koji kreira mečeve:

```
private IList<Match> CreateRoundRobinMatches(IList<User> players, int
noOfRounds)
{
    IList<Match> matches = new List<Match>();

    if (players.Count % 2 != 0)
        players.Add(new User { Id = 0 });

    int numDays = (players.Count - 1);
    int halfSize = players.Count / 2;

    List<int> playerIds = new List<int>();

    foreach (var item in players) // Copy all the elements.
        playerIds.Add(item.Id);

    playerIds.RemoveAt(0); // To exclude the first team.

    int teamsSize = playerIds.Count;

    for (int day = 0; day < numDays; day++) //Create first round of
matches
    {
        int teamIdx = day % teamsSize;

        matches.Add(new Match { Player1 = new User { Id =
playerIds[teamIdx] }, Player2 = new User { Id = players[0].Id } });

        for (int idx = 1; idx < halfSize; idx++)
        {
            int firstTeam = (day + idx) % teamsSize;
            int secondTeam = (day + teamsSize - idx) % teamsSize;

            matches.Add(new Match { Player1 = new User { Id =
playerIds[firstTeam] }, Player2 = new User { Id = playerIds[secondTeam] } });
        }

        RemoveUnwantedMatches(matches); //Removes from matches if league
has even number of players

        var allMatches = CreateRounds(matches, noOfRounds); //Creates all
other rounds for league

        return allMatches;
    }
}
```

Kôd 4.2. Kôd koji kreira mečeve

Sljedeći korak u kreiranju lige je mogućnost dodavanja još jednog administratora lige. Kreator lige je automatski postavljen kao administrator, ali aplikacija prilikom kreiranja lige dozvoljava da se nadoda još samo jedan (broj administratora je ograničen na dva), ukoliko postoji za time potreba. Dodatni administrator može biti samo onaj tko i sudjeluje u teniskoj ligi te se on može izbrati putem padajućeg izbornika u kojem se nalaze samo sudionici lige.

Zadnji korak u kreiranju lige je uređivanje pravila. Pravila određuju igrači svake lige, te ta pravila vrijede samo za njihovu ligu.

5. Korištene tehnologije

5.1. SQL

SQL je kratica za Structured Query Language. Koristi se za komunikaciju s bazom podataka. American National Standards Institute (ANSI) ga definira kao standardni jezik za sustave upravljanja relacijskim bazama podataka [18]. Naredbe se koriste kako bi se dohvatili podaci iz baze podataka ili kako bi se ti podaci ažurirali. Najpoznatiji sustavi koji koriste SQL su: Oracle, Sybase, Microsoft SQL Server, Access, Ingres...

Ovo su važne godine iz povijesti SQL-a [19]:

1970. – Dr. Edgar F. “Ted” Codd opisao je relacijski model za baze podataka.

1974. – Pojavio se Structured Query Language.

1978. – IBM je izdao proizvod pod nazivom System/R.

1986. – IBM je razvio prototip relacijske baze podataka, koju je standardizirao ANSI.

1989. – Objavljena je prva verzija SQL-a.

1999. – Objavljen je SQL 3 sa značajkama kao što su okidači, orijentacija na objekte itd.

SQL 2003 – funkcije prozora, značajke povezane s XML-om itd.

SQL 2006 – podrška za XML Query Language.

SQL 2011 - poboljšana podrška za vremenske baze podataka.

U ovom radu korištene su standardne SQL naredbe kao što su:

- **„Select“** - ova naredba koristi se kako bi se bazi podataka postavio upit i odabrali podaci koji odgovaraju navedenim kriterijima.
- **„Insert“** - ova naredba koristi se kako bi se dodao zapis u tablicu.
- **„Update“** - koristi se za promjenu zapisa u tablici.
- **„Delete“** - ova naredba koristi se za brisanje zapisa u tablici.
- **„Create“** - naredba „Create“ koristi se kako bi se stvorila nova tablica.

5.2. .NET Framework

.NET je okvir za razvoj softverskih aplikacija. Dizajnirao ga je i razvio Microsoft, a prva beta verzija izdana je 2000. godine [20].

.NET Framework se koristi za razvoj web aplikacija, aplikacija za Windows i aplikacija za mobilne uređaje. On pruža širok raspon funkcionalnosti i podrške. Također, sadrži velik broj biblioteka klasa, koje su poznate i kao Framework Class Library (FCL). Softverski programi napisani u .NET-u izvode se u izvršnom okruženju koje se naziva CLR (Common Language Runtime).

.NET Framework pruža razne usluge. Neke od njih su upravljanje memorijom, sigurnost, umrežavanje.

.NET Framework podržava više od 60 programskih jezika kao što su C#, F#, VB.NET, VC++, JScript.NET, APL, COBOL, Perl, Oberon, ML, Pascal, Eiffel, Smalltalk, Python, Cobra, itd. [20].

5.3. C#

C# (C-Sharp) je objektno orijentirani programski jezik koji je razvio Microsoft koji koristi .NET Framework [21].

Uz pomoć C# programskog jezika mogu se razviti različite aplikacije:

- Desktop aplikacije
- Web aplikacije
- Igre
- Mobilne aplikacije
- Aplikacije baze podataka...

C# je odobren kao standard od strane ECMA i ISO. Dizajniran je za CLI (engl. *Common Language Infrastructure*). CLI je specifikacija koja opisuje izvršni kôd i okruženje za izvođenje [21].

5.4. JavaScript

JavaScript je najpopularniji programski jezik [22]. Iako je najpoznatiji kao skriptni jezik za web-stranice, koriste ga i mnoga okruženja bez preglednika, poput Node.js, Apache CouchDB i Adobe Acrobat. JavaScript je dinamički jezik temeljen na prototipu, s više paradigmi, s jednom niti, koji podržava objektno orijentirane, imperativne i deklarativne (npr. funkcionalno programiranje) stilove [23].

5.5. HTML

HTML je jezik na kojem je napisana većina web stranica. Koristi se za izradu stranica i njihovu funkcionalnost [24].

HTML se upotrebljava za [25]:

- Web razvoj – koristi se HTML kôd za dizajniranje na koji način preglednik prikazuje elemente određene web stranice, kao što su tekst, hiperveze i medijske datoteke.
- Internet navigaciju – korisnici mogu na jednostavan način umetati veze između stranica i web mjesta budući da se HTML često koristi za ugrađivanje hiperveza.
- Web dokumentaciju - HTML omogućuje organiziranje i oblikovanje dokumenata, slično kao Microsoft Word.

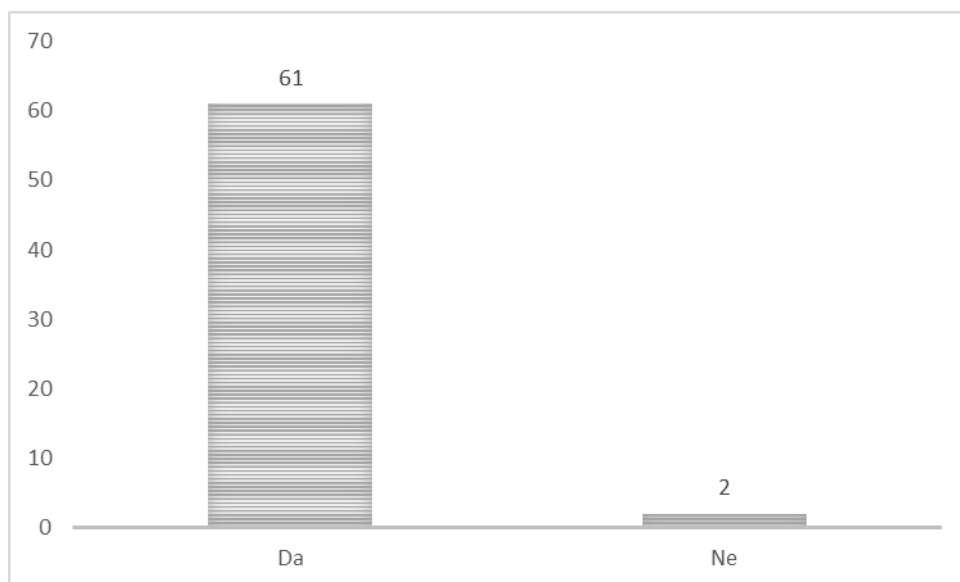
5.6. CSS

Cascading Style Sheets (CSS) je stilski jezik koji se koristi za opisivanje prezentacije dokumenta napisanog u HTML-u ili XML-u (uključujući XML dijalekte kao što su SVG ili MathML). CSS opisuje kako se elementi trebaju prikazati na ekranu, na papiru, u govoru ili na drugim medijima [26].

6. Analiza završne ankete ispitivanja zadovoljstva korisnika

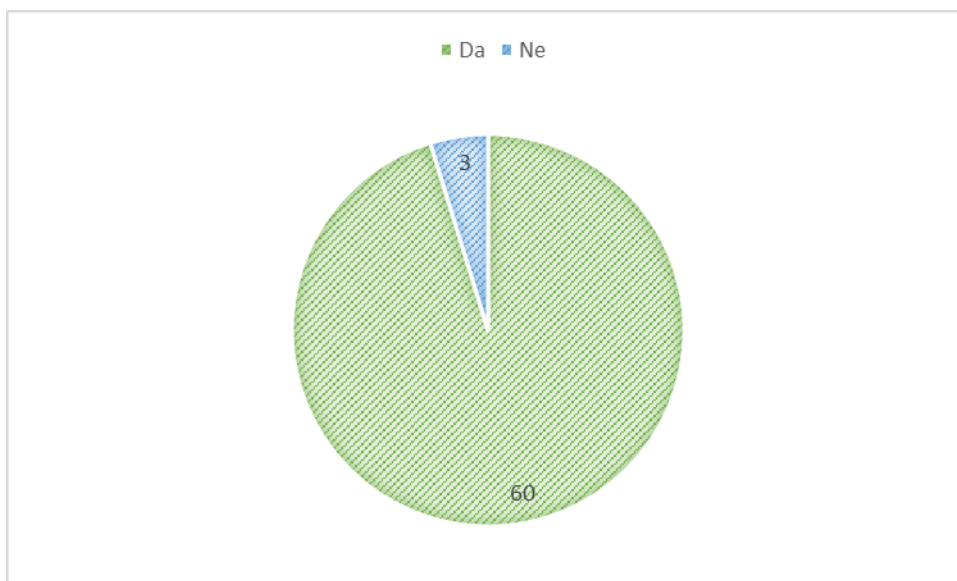
Proveden je završni anketni upitnik. U njemu je prikazano zadovoljstvo ispitanika nakon isprobavanja aplikacije. U anketnom upitniku su sudjelovala 63 ispitanika. Jednak broj ispitanika je sudjelovao u oba anketna upitnika. Anketni upitnik je proveden u vrijeme kad je aplikacija bila napravljena i funkcionalna.

Slika 6.1 prikazuje kako je 61 ispitanik (97%) odgovorio pozitivno na pitanje „Je li aplikacija ispunila Vaša očekivanja?“. Samo 2 ispitanika (3%) su odgovorila kako aplikacija nije ispunila njihova očekivanja. Iz navedenog možemo zaključiti kako je ipak većina ispitanika zadovoljna s izrađenom aplikacijom. Aplikacija je prilagođena mobilnim telefonima, tabletima i uređajima koji su u današnje vrijeme uvijek uz osobu. Nije potrebno doći do nekog stolnog ili prijenosnog računala kako bi se zabilježili rezultati.



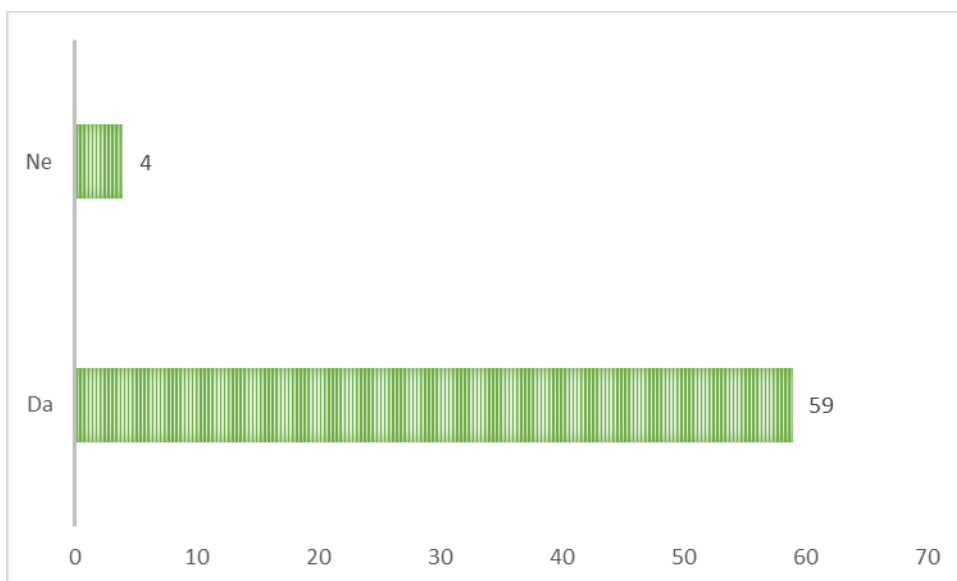
Slika 6.1. Je li aplikacija ispunila Vaša očekivanja?

Na postavljeno pitanje „Čini li Vam se korištenje aplikacije jednostavno?“ također je većina ispitanika, njih 60, odnosno 95%, odgovorila kako je korištenje aplikacije jednostavno.



Slika 6.2. Čini li Vam se korištenje aplikacije jednostavno?

Na pitanje „Sviđa li Vam se dizajn aplikacije?“, 59 ispitanika, odnosno 93% ispitanika su odgovorila kako im se sviđa dizajn. Dizajn i korištenje su vrlo jednostavni, što potvrđuju prethodna i *Slika 6.3*.



Slika 6.3. Sviđa li Vam se dizajn aplikacije?

Na pitanje „Dolazi li do nekih pogrešaka u aplikaciji?“ svi ispitanici (100%) su odgovorili kako ne dolazi do pogrešaka. Korisnici su testirali aplikaciju tri mjeseca. Prilikom testiranja aplikacije nisu uočene nikakve pogreške u njenom radu.

Posljednje pitanje u završnom anketnom upitniku je vezano za prethodno i glasi: „Ako je odgovor da, do kojih pogrešaka dolazi?“, međutim, ovdje nije dobiven niti jedan odgovor, jer su svi ispitanici na prošlom pitanju odgovorili kako nije bilo pogrešaka.

Prema provedenom anketnom upitniku može se zaključiti kako aplikacija ima lijep dizajn, jednostavna je za korištenje i ispunila je očekivanja velike većine ispitanika. Tijekom testiranja aplikacije kod ispitanika se nije pojavila niti jedna pogreška.

Zaključak

Kako su rekreacijski sportovi aktivnosti kod kojih je primarna svrha sudjelovanje, naravno i poboljšanje tjelesne spremnosti i zabava, tako je napravljena aplikacija Tennis Vision. Ideja je spajanje ljudi i poticaj na tjelesnu aktivnost. Tennis Vision je aplikacija osmišljena kako bi povezala ljude koji imaju zajednički interes, a to je u ovom slučaju tenis. Pandemija koronavirusa dodatno je utjecala na distanciranje među ljudima, što daje dodatnu motivaciju za bavljenje sportovima koji uključuju više ljudi.

Prvi anketni upitnik je pokazao kako postoji potreba za ovakvom vrstom aplikacije, što pokazuje da su ljudi željni rekreacije, druženja i zabave. Bavljenje sportom i rekreacija ne samo da imaju pozitivne fizičke učinke, nego imaju i značajan utjecaj na emocionalno zdravlje svake osobe. Drugi anketni upitnik je pokazao kako je aplikacija ispunila očekivanja ispitivača.

Prednost ove aplikacije je mogućnost širenja i daljnjeg razvijanja. Ukoliko se ukaže potrebna, aplikaciju je moguće napraviti i za druge sportove.

Drugi anketni upitnik je pokazao kako su ispitanici zadovoljni aplikacijom i njenim radom. Ova aplikacija je usmjerena prema korisnicima koji imaju slične interese, te se korisnici sličnih interesa mogu povezati.

Popis kratica

SWOT	<i>Strengths, Weaknesses, Opportunities, Threats</i>	snage, slabosti, prilike, prijetnje
API	<i>Application Programming Interface</i>	sučelje za programiranje aplikacija
SQL	<i>Structured Query Language</i>	strukturirani jezik upita
ER	<i>Entity relationship</i>	odnos entiteta
CLI	<i>Common Language Infrastructure</i>	zajednička jezična infrastruktura
MVC	<i>Model View Controller</i>	kontroler prikaza modela
SVG	<i>Scalable Vector Graphics</i>	skalabilna vektorska grafika
HTML	<i>HyperText Markup Language</i>	jezik za označavanje hiperteksta
CSS	<i>Cascading Style Sheets</i>	listovi kaskadnog stila
XML	<i>Extensible Markup Language</i>	jezik za označavanje podataka
C#	<i>C-Sharp</i>	programski jezik
F#	<i>F-Sharp</i>	programski jezik
COBOL	<i>Common Business Oriented Language</i>	programski jezik
ECMA	<i>European Computer Manufacturers Association</i> računala	udruženje europskih proizvođača računala
ISO	<i>International Organization for Standardization</i> normizaciju	međunarodna organizacija za normizaciju
XML	<i>Extensible Markup Language</i>	jezik za označavanje podataka

Popis slika

Slika 2.1. Postotak stanovništva s prekomjernom tjelesnom težinom u EU, prema dobi	3
Slika 2.2. Postotak stanovništva s prekomjernom tjelesnom težinom po državama	4
Slika 3.1. SWOT analiza	7
Slika 3.2. Spol ispitanika	8
Slika 3.3. Dob ispitanika	9
Slika 3.4. Igrate li tenis rekreativno?	9
Slika 3.5. Sudjelujete li u ikakvim amaterskim natjecanjima?	10
Slika 3.6. Bilježite li rezultate odigranih mečeva u bilo kojem obliku?	10
Slika 3.7. Bilježite li rezultate odigranih mečeva u bilo kojem obliku?	11
Slika 3.8. Bi li Vas zanimala aplikacija za organizaciju vlastite lige te praćenje rezultata svojih odigranih mečeva?	11
Slika 4.1. Troslojna arhitektura aplikacije	13
Slika 4.2. Dijagram baze podataka	16
Slika 4.3. Servisni sloj	17
Slika 4.4. MVC arhitektura	18
Slika 4.5. Početni ekran Tennis Vision	19
Slika 4.6. Prijava/registracija	19
Slika 4.7. Prikaz stranice nakon uspješne prijave u aplikaciju	21
Slika 6.1. Je li aplikacija ispunila Vaša očekivanja?	27
Slika 6.2. Čini li Vam se korištenje aplikacije jednostavno?	28
Slika 6.3. Sviđa li Vam se dizajn aplikacije?	28

Popis tablica

Tablica 3.1. Usporedba konkurencije.....	8
Tablica 4.1 Tablice u bazi podataka.....	15

Popis kôdova

Kôd 4.1. Kôd za smanjivanje slika.....	20
Kôd 4.2. Kôd koji kreira mečeve	22

Literatura

- [1] Latković, I., Rukavina, M., Schuster, S. *Učestalost Ozljeda I Kvaliteta Života Kod Rekreativnog Bavljnja Sportom*. Zdravstveno veleučilište, Zagreb, Hrvat. Športskomed. Vjesn. 2015; 30: 98-105
- [2] Cox, R. *Psihologija Sporta: Koncepti Primjene*. Jastrebarsko: Naklada Slap, 2005.
- [3] Milanović, D, Jukić, I, Čustonja, Z, Šimek, S. *Tjelesno Vježbanje I Sport U Odgojno Obrazovnom Procesu – Strateški Pristup*. U: Prskalo I, ur. Zbornik radova 5. dani Mate Demarina, Petrinja: Visoka učiteljska škola. 2004; 28 – 37.
- [4] <https://repositorij.mef.unizg.hr/islandora/object/mef%3A3321/datastream/PDF/view>, 01.02.2023.
- [5] <https://ec.europa.eu/eurostat/web/products-eurostat-news/-/ddn-20210721-2>, 28.09.2022.
- [6] Galić, M., The best croatian male tennis players at a turning point – studies or professional sport. University of Zagreb Faculty of Political Science, Croatia, 9th International Scientific Conference on Kinesiology, 2021, Opatija, Croatia
- [7] <https://study.com/learn/lesson/tennis-history-background.html>, 30.09.2022.
- [8] <https://ferivisport.hr/blog/post/tenis-kao-popularan-rekreativni-sport1>, 10.10.2022.
- [9] <https://hr.thpanorama.com/articles/administracin-y-finanzas/anlisis-situacional-en-qu-consiste-cmo-se-hace-y-ejemplo.html>, 10.10.2022.
- [10] <https://hr.thpanorama.com/articles/administracin-y-finanzas/anlisis-situacional-en-qu-consiste-cmo-se-hace-y-ejemplo.html>, 20.10.2022.
- [11]] <https://www.investopedia.com/terms/s/swot.asp>, 20.10.2022.
- [12] <https://tenisliga.com/Pages2/Default.aspx>, 28.10.2022.
- [13] <https://localtennisleagues.com/>, 28.10.2022.
- [14] <https://oztennisleagues.com/>, 05.11.2022.
- [15] <https://www.techopedia.com/definition/24649/three-tier-architecture>, 14.12.2022.
- [16] <https://www.tutorialsteacher.com/mvc/mvc-architecture>, 14.12.2022.
- [17] <https://www.techtarget.com/whatis/definition/bootstrap>, 02.02.2023.
- [18] <https://www.sqlcourse.com/beginner-course/what-is-sql/>, 14.12.2022.
- [19] <https://www.guru99.com/what-is-sql.html>, 14.12.2022.
- [20] <https://www.javatpoint.com/net-framework>, 18.12.2022.
- [21] <https://www.javatpoint.com/c-sharp-tutorial>, 18.12.2022.
- [22] <https://javascript.plainenglish.io/why-javascript-is-the-worlds-most-popular-programming-language-4da6b793568d>, 18.12.2022.

[23] <https://developer.mozilla.org/en-US/docs/Web/JavaScript>, 27.12.2022.

[24] https://html.com/#What_is_HTML, 27.12.2023.

[25] <https://www.hostinger.com/tutorials/what-is-html>, 02.02.2023.

[26] <https://developer.mozilla.org/en-US/docs/Web/CSS>, 27.12.2022.