

Web portal za kolekcionarsku kartašku igru

Vuko, Marin

Undergraduate thesis / Završni rad

2023

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:200:950015>

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

Download date / Datum preuzimanja: **2024-07-13**

Repository / Repozitorij:

[Faculty of Electrical Engineering, Computer Science and Information Technology Osijek](#)



**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I
INFORMACIJSKIH TEHNOLOGIJA**

Stručni studij računarstva

**WEB PORTAL ZA KOLEKCIONARSKU KARTAŠKU
IGRU**

Završni rad

Marin Vuko

Osijek, 2023.

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA
I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK****Obrazac Z1S: Obrazac za imenovanje Povjerenstva za završni ispit na preddiplomskom stručnom studiju**

Osijek, 07.07.2023.

Odboru za završne i diplomske ispite

**Imenovanje Povjerenstva za završni ispit
na preddiplomskom stručnom studiju**

Ime i prezime Pristupnika:	Marin Vuko
Studij, smjer:	Preddiplomski stručni studij Računarstvo
Mat. br. Pristupnika, godina upisa:	AR 4765, 19.07.2019.
OIB Pristupnika:	21222746748
Mentor:	Robert Šojo, mag. ing. comp.
Sumentor:	,
Sumentor iz tvrtke:	
Predsjednik Povjerenstva:	Marina Peko, dipl. ing.
Član Povjerenstva 1:	Robert Šojo, mag. ing. comp.
Član Povjerenstva 2:	dr. sc. Ivana Hartmann-Tolić
Naslov završnog rada:	Web portal za kolekcionarsku kartašku igru
Znanstvena grana završnog rada:	Programsko inženjerstvo (zn. polje računarstvo)
Zadatak završnog rada	Web portal sadrži bazu informacija o kartama kartaške igre, vođenje turnira. Vođenje portala obrađuje administrator koji dodaje karte u bazu sa svim njihovim informacijama. Administrator ima uvid u analitiku igrača, karata i turnira. Ideja portala je omogućiti registraciju korisnika koji će sudjelovati u online turnirima, odabir karata, izrada svog tima karata. Omogućiti turnire s drugim timovima te napraviti funkcionalnu virtualni dvoboj. Student treba razviti funkcionalnu aplikaciju koristeći različite web tehnologije. Rezervirano za studenta: Marin Vuko
Prijedlog ocjene pismenog dijela ispita (završnog rada):	Izvrstan (5)
Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:	Primjena znanja stečenih na fakultetu: 3 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 3 bod/boda Jasnoća pismenog izražavanja: 3 bod/boda Razina samostalnosti: 3 razina
Datum prijedloga ocjene od strane mentora:	07.07.2023.
<i>Potvrda mentora o predaji konačne verzije rada:</i>	<i>Mentor elektronički potpisao predaju konačne verzije.</i>
	Datum:

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA
I INFORMACIJSKIH TEHNOLOGIJA OSIJEK**IZJAVA O ORIGINALNOSTI RADA**

Osijek, 15.07.2023.

Ime i prezime studenta:

Marin Vuko

Studij:

Preddiplomski stručni studij Računarstvo

Mat. br. studenta, godina upisa:

AR 4765, 19.07.2019.

Turnitin podudaranje [%]:

10

Ovom izjavom izjavljujem da je rad pod nazivom: **Web portal za kolekcionarsku kartašku igru**

izrađen pod vodstvom mentora Robert Šojo, mag. ing. comp.

i sumentora ,

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija. Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

SADRŽAJ

1. UVOD	1
1.1. Zadatak završnog rada	2
2. POSTOJEĆE APLIKACIJE	3
2.1. Bulbapedia	3
2.2. Serebii.net.....	5
2.3. Mariland.....	6
3. KORIŠTENE TEHNOLOGIJE PRI IZRADU WEB PORTALA	7
3.1. Visual Studio Code	8
3.2. HTML.....	8
3.3. JavaScript	10
3.4. CSS.....	12
3.5. Bootstrap	14
3.6. Ajax.....	15
3.7. SQL.....	16
3.8. PHP	17
4. IZRADA APLIKACIJE	19
4.1. Baza podatka	19
4.2. Početna stranica.....	22
4.3. Pokédex	24
4.4. Prijava i registracija.....	26
4.5. Izgradnja tima	28
4.6. Chat	30
4.7. Upravljača ploča.....	31
4.8. Turnir i ljestvica	32

5. KORIŠTENJE APLIKACIJE	39
5.1. Neregistrirani korisnik	40
5.2. Registrirani korisnik	44
5.3. Admin i vlasnik.....	50
6. Zaključak	52
LITERATURA	53
SAŽETAK.....	56
ABSTRACT	57

1. UVOD

U posljednja dva desetljeća internet se počeo ubrzano širiti, te je do sada zahvatio sve aspekte naših života. Biti povezan sa svim ljudima na mreži je ubrzo postala potpuno normalna stvar, ljudi su dolazili na internet kako bi pronašli vijesti koje ih zanimaju, recepte za jela, razgovore s ljudima s drugoga kraja svijeta, te isto tako kako bi pronašli ljude koji dijele slične interese s njima. Objavom svojih dviju igrica jedna od najvećih franšiza svijeta „Pokémon“ je 1996. godine zaintrigirala jako velik broj ljudi u svijetu, ali tada nije bilo interneta kako ga mi danas znamo, te ljudi tada nisu imali pristup svim informacijama o igricama i ljudima s kojima bih mogli pričati o igrici. Danas je stanje po pitanju „Pokémon“ obožavatelja skroz drugačije, postoje razne stranice i udruge u kojima se ljudi na internetu mogu povezati s ostalim ljubiteljima franšize, te podijeliti svoje iskustvo, osjećaje i ideje s drugima. Na web portalu se nalaze razne funkcionalnosti, korisnici mogu vidjeti sve pokemone u pokedexu nebitno jesu li prijavljeni ili ne, mogu čitati o svakoj generaciji pokemona i njihovim konzolama, postoji opcija za izgradnju vlastitog tima od prve generacije, komunikacija s ostalim korisnicima u vidu chata, te vođenje turnira između osam igrača.

U drugom poglavlju napravljen je pregled stranica koje nude slična rješenja kao ovaj web portal. Svaka od njih je jedinstvena, te pruža malo drugačiji pristup svim informacijama vezano za svijet pokemona.

U trećem poglavlju opisane su tehnologije koje se koriste za razvoj ove web aplikacije. Sve tehnologije koje su opisane su korištene kako bi se na kraju došlo do funkcionalne web aplikacije, opisano je sve od toga što je korišteno za stvaranje bazne podatka, spajanje na nju, te ostale tehnologije koje su korištene za oblikovanje prikaza i stvaranje funkcionalnosti u pozadini.

U četvrtom poglavlju detaljno je opisan postupak razvoja aplikacije. Prikazan je kod koji se koristi za izradu određenih dijelova web aplikacije. Vidljivo je sve od stvaranja baze podataka do vođenja turnira.

U petom poglavlju je prikazano kako se web aplikacija koristi. Prvo je prikazano što je sve dostupno neregistriranom korisniku, nakon toga prikazane su sve opcije koje su dostupne samo registriranim korisnicima, te čemu još dodatno admin i vlasnik imaju pristup.

1.1. Zadatak završnog rada

Zadatak završnog rada je razviti web aplikaciju koja će na jednom mjestu korisnicima pružati sve informacije vezane uz „Pokémon“ franšizu. Na stranici se za ne prijavljene korisnike nalaze poveznice na vanjske stranice, stranice o svakoj generaciji te o konzoli na kojoj su se igrale, „Pokédex“ s informacijama do kraja 8. generacije. Prijavljeni korisnici uz sve već navedeno imaju pristup chatu u kojemu mogu komunicirati s ostalim prijavljenim korisnicima. Isto tako imaju opciju izgradnje tima od 3 pokemona iz prve generacije, te opciju turnira u kojemu se dodaje osam igrača koji igraju uživo te prate napredak na stranici i tko ima najviše bodova na ljestvici. Korisnici koji imaju titulu admina imaju pristupu radnoj ploči na kojoj su vidljivi svi korisnici, te ih oni mogu obrisati, dok postoji samo jedan „owner“ stranice koji može uz brisanje korisnika dodati i maknuti titulu admina.

2. POSTOJEĆE APLIKACIJE

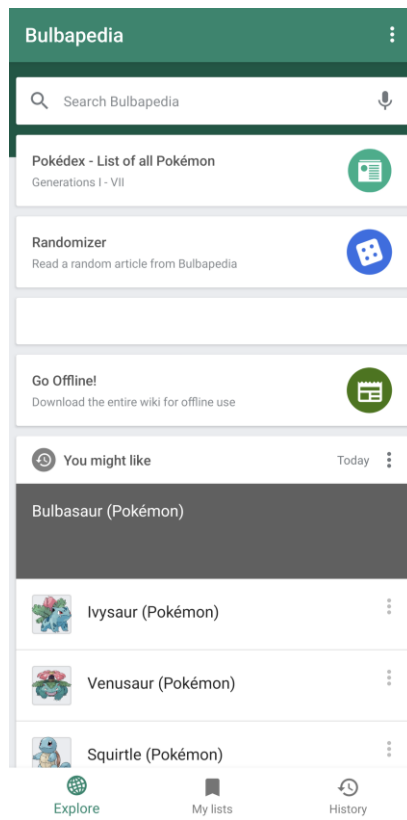
U ovom poglavlju se prikazuje aplikacije koje rade slične stvari kao i završni rad.

2.1. Bulbapedia

Bulbapedia je otvorena enciklopedija o Pokémonima, kojoj svatko može pridonijeti. Osnovana je 2006. godine i od tada je postala jedna od najvećih stranica na internetu za novosti i sve vezano uz Pokémone. Trenutno imaju 46 tisuća članaka. Izgled početne stranice za web se može vidjeti na slici 2.1 , a na slici 2.2 se može vidjeti početna stranica mobilne aplikacije [1].



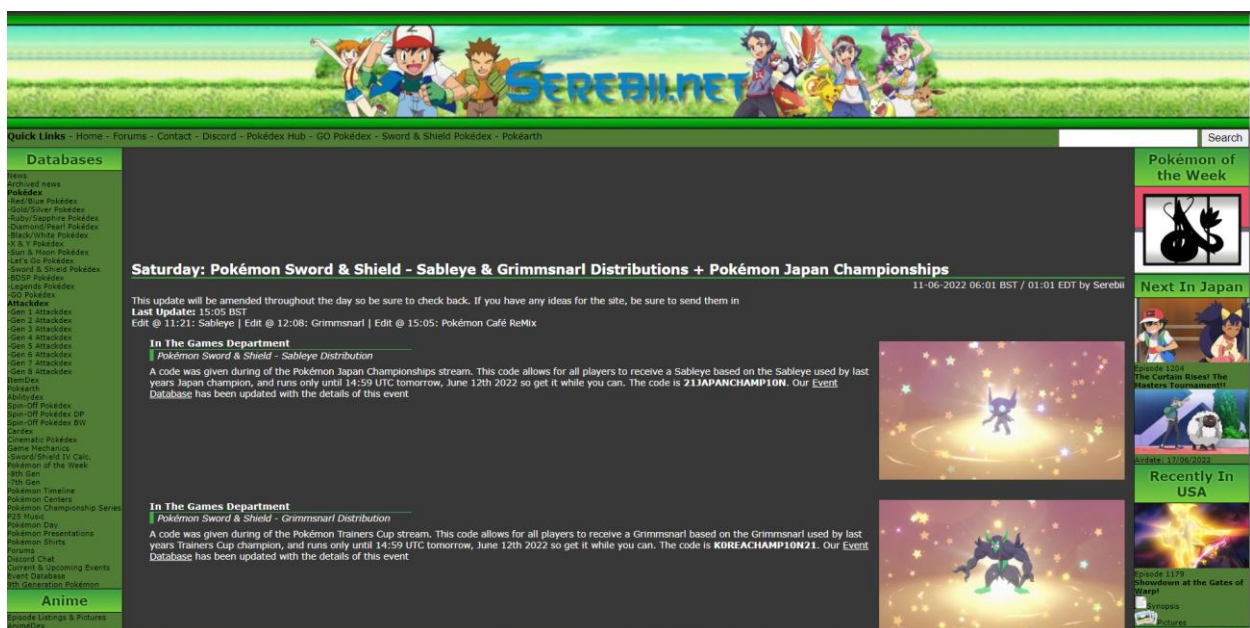
Sl. 2.1. Izgled početne stranice Bulbapedia na web-u



Sl. 2.2. Izgled početne stranice Bulbapedia na mobitelu

2.2. Serebii.net

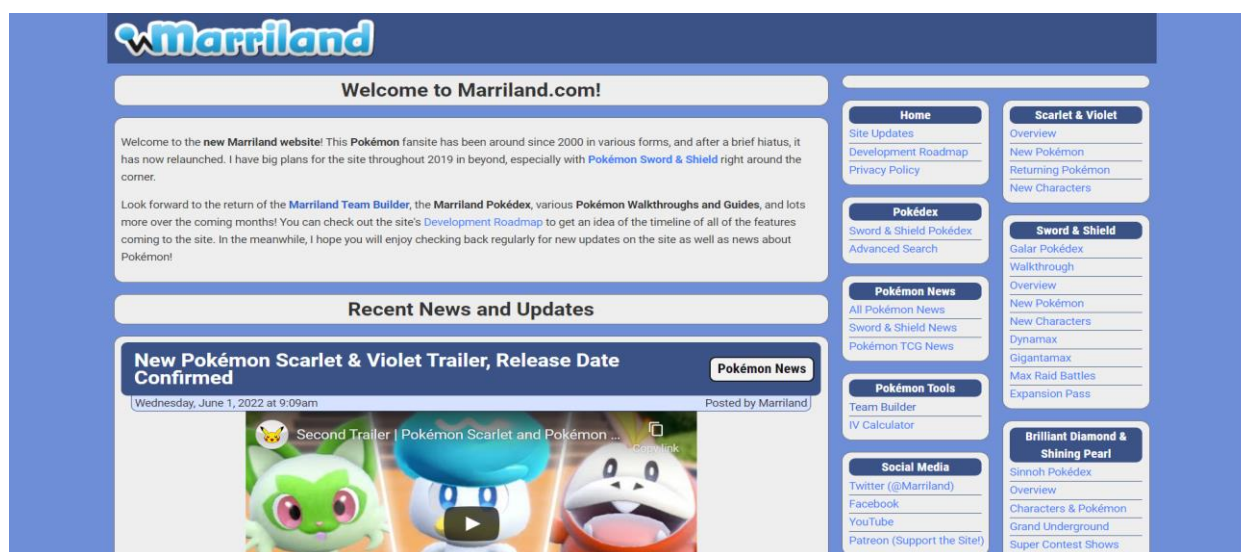
Serebii.net je stranica koju je napravio Joe Merrick, stranica je napravljena 1999.godine kada su Pokémoni krenuli skupljati pozornost javnosti. Jedina osoba koja ažurira stranicu je ujedno i njen vlasnik, iako stranica ima samo jednu osobu koja radi na njoj konstantno je u toku sa svim vijestima. Stranica sadrži jedinstvenu kartu nazvanu „Pokéarth“ koja ima sve trenere i njihove Pokémone, i na kojim mjestima se nalaze koji Pokémoni. Početna web stranica se može vidjeti na slici 2.3 [2].



Sl. 2.3. Izgled početne stranice Serebii.net na web-u

2.3. Mariland

Mariland je web stranica koja u ovoj verziji novija od prethodne dvije. Stranica je nakon godina neaktivnosti ponovno pokrenuta u novom izdanju 2019.godine. Na stranici se nalaze sve novosti vezane za novije Pokémon igre. Nedostatak stranice je što ne pokriva toliki raspon vijesti kao prethode dvije, ali prednost je što ima opciju za izgradnju svoga tima, gdje stranica odmah govori je li Pokémon dostupan u određenim igricama. Početna stranica se može vidjeti na slici 2.4 [3].



Sl. 2.4. Izgled početne stranice Merriland na web

3. KORIŠTENE TEHNOLOGIJE PRI IZRADU WEB PORTALA

Za razvoj web aplikacija postoji jako veliki izbor programskih tehnologija i jezika koji služe za pisanje aplikacije. Neke od tehnologija za izradu web aplikacije se mogu vidjeti na slici 3.1 [4].

Programske tehnologije se dijele na backend i frontend tehnologije.

Frontend tehnologije služe kako bi se prikazao vizualni dio aplikacije, kao npr. tekst, gumb i navigacija. Kada web dizajneri naprave skicu izgleda stranice, frontend programeri ju moraju napisati u kodu kako bi bila funkcionalna. Frontend je primjer statičke web stranice, to znači da se sadržaj na stranici ne mijenja puno.

Backend tehnologije služe za sve što korisnik ne vidi na web stranici, kao što je baza podataka. Postoji konstanta komunikacija s frontendom jer kada korisnik preko preglednika pošalje zahtjev, backend taj zahtjev obrađuje i vraća odgovarajuću stvar korisniku. Backend je primjer dinamičke web stranice, to znači da se u stvarnome vremenu ona ažurira.



Sl. 2.1. Tehnologije korištene za razvoj web aplikacija

3.1. Visual Studio Code

Visual studio Code ili samo VS Code je uređivač koda koji je napravila tvrtka Microsoft 2015. godine. Podržava preko 100 programskih jezika. Program je napravljen u TypeScriptu, JavaScriptu, HTML-u (engl. *HyperText Markup Language*) i CSS-u (engl. *Cascading Style Sheets*), namijenjen je za Windows, Linux i MacOS operacijske sustave. Neke od značajki su podrška za debugging, snippeti, refaktoriranje koda i ugrađeni Git. U 2021. godini Stack Overflow je održao „Developer Survey“ i VS Code je bio najpopularniji razvojni alat za čak 70% sudionika [4].

3.2. HTML

HTML je jezik koji se koristi za izradu web stranica. HTML nije programski jezik, nego služi za opis hipertekstualnih datoteka. Jezik ne može izvršavati računalne operacije i izvoditi petlje, zato mu kao asistent služi JavaScript kako bi se preko njega odradila određena operacija, isto tako kao asistent služi i CSS koji definira izgled i raspored sadržaja na stranici. Web preglednik preko web servera ili lokalne baze dobiva HTML dokumente i renderira ih u web stranicu.

Osnivač HTML-a je Tim Berners-Lee koji je osnovao jezik 1991.godine. Ideja koja je potaknula na razvoj HTML-a je bila ideja internetskog hiperteksta, to je tekst koji sadrži poveznice na druge tekstove kojima korisnici mogu odmah pristupiti, a autor može slati svoje korisnike na druge radove kako ih ne bi morao citirati. Prva verzija HTML 1.0 je objavljena tek 1993. godine, sastojala se od 18 oznaka, dok trenutno ima oko 140 oznaka. Druga verzija HTML 2.0 je objavljena 1995.godine i sadržavala je sve funkcije HTML 1.0 uz još dodatne funkcije. Treća verzija HTML 3.0 je objavljena 1997.godine, u ovoj verziji je Dave Ragget uveo nove funkcije HTML koje su bile odlične za veću slobodu dizajniranja, ali te promjene su usporile rad web preglednika. Četvrta verzija HTML 4.0 izlazi 1999.godine, nadograđena četvrta verzija HTML 4.01 izlazi 2012.godine, dok trenutno zadnja verzija HTML 5.0 je izašla 2014.godine. Iz slike 3.2 može se vidjeti kako je HTML s vremenom uznapredovao kako bi prikazao više stvari na web stranicama.

Na slici 3.3 vidi se razlika između elementa, atributa i oznake u HTML-u. Svaki dokument mora imati tri glavne oznake, a to su <html>, <head> i <body> [5].

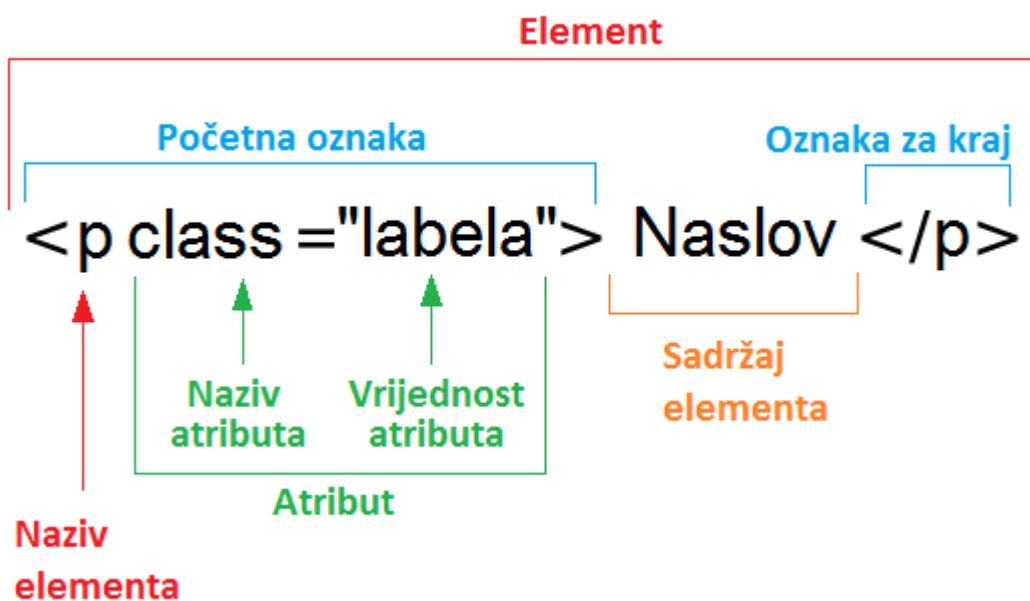
<html> označava početak dokumenta.

<head> označava zaglavlje, u koji nam ide naslov stranice, stil i definiranje skripti.

<body> označava sav sadržaj koji se nalazi na stranici.

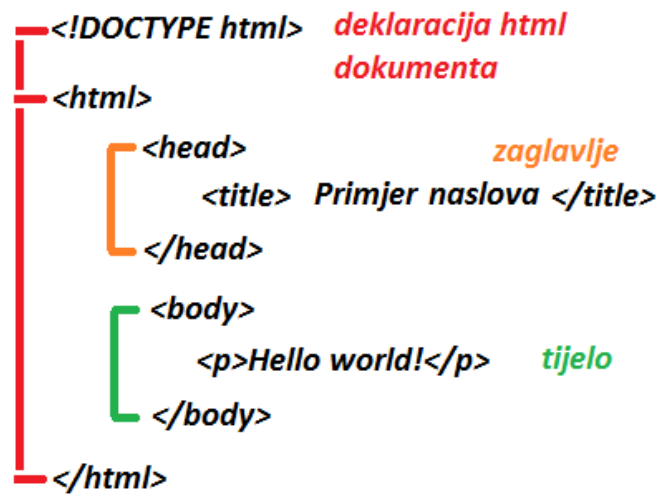
Sadržaj	HTML 1.2	HTML 4.01	HTML 5	Svrha
Zaglavlje	Da	Da	Da	Organizira sadržaj stranice tako da dodaje zaglavlja i podzaglavlja na vrh svakog dijela stranice
Odlomak	Da	Da	Da	Prepoznavanje određenog odlomka teksta
Adresa	Da	Da	Da	Prepoznaje odlomak teksta koji sadrži podatke o kontaktu
Sidro	Da	Da	Da	Poveznica na druge sadržaje na web-u
Lista	Da	Da	Da	Organizira stavke u listu
Slika	Da	Da	Da	Ugrađivanje slike ili crteža u web stranicu
Tablica	Ne	Da	Da	Organizira podatke u redove i stupce
Stil	Ne	Da	Da	Dodavanje CSS-a za kontrolu kako su predmeti na web stranici raspoređeni
Skripta	Ne	Da	Da	Dodavanje JavaScript-a na stranicu kako bi bila responzivna na korisničko ponašanje
Zvuk	Ne	Ne	Da	Dodavanje zvuka na stranicu s jednom oznakom
Video	Ne	Ne	Da	Dodavanje videa na stranicu jednom oznakom

Sl. 3.2. Razlike između HTML verzija



Sl. 3.3. tech

Raspored oznaka u kodu je vidljiv na slici 3.4 [5].



Sl. 3.4. Glavne oznake u kodu

3.3. JavaScript

JavaScript često samo JS je skriptni jezik koji omogućuje implementaciju kompleksnih značajki na web stranice, neke od značajki mogu biti dinamičko ažuriranje sadržaja, kontroliranje multimedije, animiranje slika i akcije na pritisak gumba. JavaScript je jedna od temeljnih tehnologija za WWW, zajedno s HTML-om i CSS-om, čak 98.7% stranica koriste JavaScript [7].

Deklaracija varijabli se vrši uz ključnu riječ `var`, npr: `var imeStudenta`, varijable se dijele na cjelobrojne, decimalne, znakovne i logičke.

Najčešći slučajevi za korištenje JavaScripta:

1. Dodavanje interaktivnog ponašanja aplikaciji
2. Razvijanje web i mobilnih aplikacija
3. Razvoj jednostavnih web servera
4. Razvoj igrica

JavaScript služi kako bi u HTML uveli nekakvu funkcionalnost, mogu se uvesti petlje, ali isto tako mogu se uvesti radnje koje bi se izvršile nakon korisnikove interakcije s aplikacijom (okidači) npr. klik na gumb izvrši nekakvu operaciju.

JavaScript kod se može koristiti u HTML-u na 2 načina:

1. Spremanje u JavaScript datoteku s proširenjem .js, te ga pozvati u <script> oznakama, to možemo vidjeti na slici 3.5 .

2. Spremanje u <head> ili <body> pod <script> oznakama, to možemo vidjeti na slici 3.6

<i>Linija</i>	<i>Kod</i>
1:	<body>
2:	<script type="text/javascript">
3:	var i = 0;
4:	var n = 0;
5:	while(i < 5){
6:	i++;
7:	if(i == 3){
8:	continue;
9:	}
10:	n += i;
11:	Console.log(n); //ispis će biti 1,3,7,12
12:	}
13:	</script>
14:	</body>

Sl. 3.5. Korištenje while petlje u HTML-u pomoću JavaScripta

<i>Linija</i>	<i>Kod</i>
1:	<head>
2:	<script type="text/javascript" src="/file.js"> </script>
3:	</head>

Sl. 3.6. Pozivanje eksterne JavaScript datoteke

3.4. CSS

CSS je stilski jezik koji služi kako bi se oblikovao sadržaj dokumenta koji je napisan u HTML-u. Može se koristiti za animacije, promjenu boje teksta, margine, raspored stvari na stranici, uređivanje linkova i slično. Zajedno s HTML-om i JavaScriptom je jedan od temeljnih tehnologija World Wide Weba.

CSS je predložen 1994.godine, a napravili su ga Håkon Wium Lie i Tim Berners-Lee dok su radili u CERN-u. Prva verzija je objavljena 1996.godine, druga verzija 1998.godine a nakon CSS 2.1 svaki modul se može samostalno razvijati, posljednja verzija je CSS3.

CSS sintaksa se može vidjeti na slici 3.7, sastoji od tri dijela:

1. Selektori – govori kojem elementu u HTML-u će se pridružiti određena pravila
2. Svojstva – to su karakteristike koje opisuju određene stvari npr. color i background
3. Vrijednosti - to je vrijednost koje određeno svojstvo prima npr. black za color svojstvo

<i>Linija</i>	<i>Kod</i>
1:	h1{ /*selektor*/
2:	color: red; /*color je svojstvo, red je vrijednost*/
3:	} /*komentar se ovako pise*/

Sl.3.7. CSS sintaksa

Postoje tri načina kako koristiti CSS u HTML datoteci:

1. Linijski – koristi se ključna riječ style u HTML elementima, isječak koda 3.8
2. Unutarnji – koristi se <style> element u <head>, isječak koda 3.9
3. Vanjski – koristi se <link> kako se implementirala vanjska css datoteka, isječak koda 3.10

<i>Linija</i>	<i>Kod</i>
1:	<body>
2:	<h1 style="color:blue"> Naslov </h1>
3:	</body>

Sl. 3.8. Linijska implementacija

<i>Linija</i>	<i>Kod</i>
1:	<head>
2:	<style>
3:	h1{
4:	color: red;
5:	font-family: verdana;
6:	Font-size: 150%;
7:	}
8:	</style>
9:	</head>

Sl. 3.9. Unutarnja implementacija

<i>Linija</i>	<i>Kod</i>
1:	<head>
2:	<link rel="stylesheet" href="styles.css">
3:	</head>

Sl. 3.10. Vanjska implementacija

Postoji pet vrsta CSS selektora:

1. Jednostavni – neki od njih su: #, . i *
2. Kombinatorski – neki od njih su >, + i ~
3. Pseudo-klasni – neki od njih su: :visited, :hover i :active
4. Pseudo-elementi – neki od njih su: ::first-line, ::first-letter, ::before i ::after
5. Atributni – neki od njih su: [target], [target="_blank"] i [title="FERIT"]

3.5. Bootstrap

Bootstrap je najpopularniji besplatni frontend CSS okvir (engl. *Framework*) otvorenog izvornog koda, svrha programa je usmjerena prema responzivnom dizajnu koji je primarno orijentiran prema mobilnim uređajima. Sastoji se od HTML-a, CSS-a, i JS biblioteka, implementira se u <head> kod HTML-a. Omogućava programeru uređivanje sadržaja svoje stranice u toj liniji koda na koju se odnosi, nudi jako veliku fleksibilnost, ima već ugrađene predloške za formu, tipografiju, gumbе, tablice, navigaciju, modale, slike i još puno toga. Primjer implementacija bootstrapa se može vidjeti na slici 3.11.

Linija	Kod
1:	<head>
2:	<link rel="stylesheet"
3:	href= https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css >
4:	<script
5:	src= https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js >
6:	</script>
7:	<script
8:	src= https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js.bootstrap.min.js >
9:	</script>
10:	</head>
11:	<body>
12:	<div class="container-fluid">
13:	<h1>Bootstrap stranica</h1>
14:	<p>Tekst</p>
15:	</div>
16:	</body>

Sl. 3.11. Implementacija Bootstrapa u HTML

Bootstrap se originalno zvao Twitter Blueprint i razvili su ga Mark Otto i Jacob Thornton, ideja koja ih potaknula na razvoj je bila ta da je postojalo jako puno biblioteka za uređivanje sučelja i to je stvaralo problem u obliku nedosljednosti, te su odlučili stvoriti novi okvir kako bi ih sve objedinili. Projekt je prvi puta objavljen 2011.godine, zadnja verzija je Bootstrap 5 koja je objavljena 2021.godine i uvela je jako puno novih funkcionalnosti.

3.6. Ajax

AJAX je skraćenica od (engl. *Asynchronous JavaScript and XML*), i nije programski jezik, nego je to grupa raznih tehnologija. Koristi se zajedno s HTML-om i CSS-om. Aplikacije koje imaju AJAX se učitavaju samo jednom, a komunikacija sa serverom se odvija asinkrono, podaci se razmjenjuju u pozadini bez da ometaju klijentovo iskustvo. Omogućeno je da se samo dio stranice ažurira s novim podacima, umjesto da se cijela stranica mora osvježiti kako bi prikazala promjene, što poboljšava rad stranice. Bez AJAX-a stranice su se učitavale sinkronom metodom, tj. da se prilikom izvršavanja nekog zahtjeva stranica morala cijela ponovno učitati od početka do kraja.

Prije 2005.godine bilo je jako teško definirati vezu između klijenta i servera, ali 2005.godine James Garrett je napisao uvodnik po imenu „AJAX: A replacement Approach to Web Applications“.

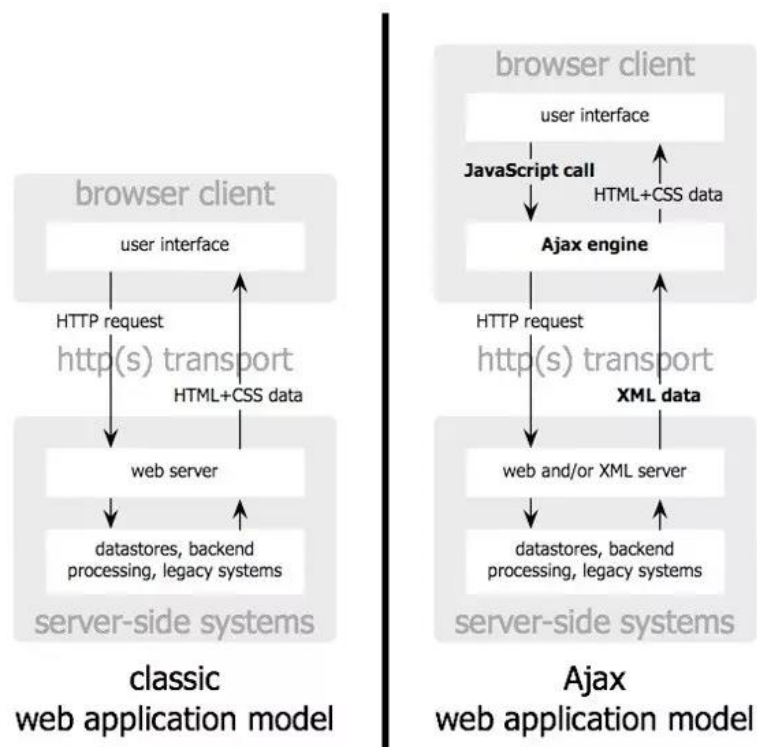
Koraci klasičnog modela (slika 3.12 lijevi dio) [6]:

1. Na korisnikov zahtjev HTTP zahtjev se šalje iz preglednika prema poslužitelju, korisnik ne vidi promjene dok se zahtjev ne obradi
2. Zahtjev dolazi do poslužitelja, i on dohvaća potrebne podatke
3. Zatraženi podaci se vraćaju u preglednik, stranica se osvježava i korisnik ih može vidjeti

Koraci AJAX modela (slika 3.12 desni dio) [6]:

1. Korisnik obavi nekakvu radnju i preglednik izvodi JavaScript stvara XMLHttpRequest objekt, koji služi za komunikaciju sa serverom.
2. Zahtjev dolazi do poslužitelja i dohvaća potrebne podatke

3. JavaScript, XML i HTML podaci se vraćaju AJAX-ovom mehanizmu, i on ih onda šalje nazad u preglednik



Sl. 3.7. Rad AJAX-a

3.7. SQL

SQL je skraćenica od (engl. *Structured Query Language*), to je jezik za baze podataka, zadužen je za dohvaćanje i organiziranje podatka u relacijskoj bazi. Može definirati baze podataka, prava pristupa, ER dijagrame (engl. *Entity Relationship*), administracija korisnik i sl.

Razvili su ga Donald D.Chamberlin i Raymond F.Boyce u IBM-u, u ranim 1970-ima. Verzija se zvala SEQUEL (engl. *Structured English Query Language*) i služila je za manipulaciju i dohvaćanje podatka iz IBM-ove tadašnje System R baze podatka.

SQL baze većinom imaju jednu ili više tablica, svaka tablica mora imati svoje definirano ime, i svaka tablica sadrži redove u kojima se spremaju podaci.

Najbitnije SQL komande:

1. SELECT – dohvaća podatke iz baze
2. UPDATE – ažurira podatke u bazi

3. DELETE – briše podatke iz baze
4. INSERT INTO – postavlja nove podatke u bazu
5. CREATE DATABASE – stvara novu bazu podataka
6. ALTER DATABASE – modificira bazu podataka
7. CREATE TABLE – kreira novu tablicu
8. ALTER TABLE – modificira tablicu
9. DROP TABLE – briše tablicu
10. CREATE INDEX – stvara index
11. DROP INDEX – briše index

Elementi SQL jezika:

1. Ključne riječi – svaki izraz ih sadržava npr. SELECT i FROM.
2. Identifikatori – imena objekata u bazi npr. Ime, Prezime, Spol.
3. Stringovi – doslovni stringovi ili CHAR i VARCHAR tipovi podataka.
4. Izrazi – stvaraju se od različitih vrsta elemenata, npr. konstanti, imena, operatora i sl.
5. Uvjeti pretraživanja – ključne riječi su WHERE, HAVING, CHECK, JOIN, IF.
6. Specijalne vrijednosti – mogu se koristiti u izrazima npr. CURRENT DATE
Specijalna Vrijednost vraća trenutnu godinu, mjesec i dan.
7. Varijable – imamo 3 vrste a to su lokalne, globalne i varijable na razini veze.
8. Komentari – služe za objašnjenje koda, preporučuje se korištenje /* */, ali može se koristiti i --, // i %.
9. NULL vrijednost – koristi se za specificiranje vrijednosti kojoj je vrijednost nepoznata.

MySQLi znači (engl. *MySQL improved*), objektno je orijentiran i najviše se koristi u PHP-u. U suštini MySQLi nije baza podataka nego poboljšani interface za prikazivanje MySQL baze podataka. Ima ugrađeni mehanizam koji sprječava SQL ubacivanje. Objavljen je u više paketa koji su bili pušteni od 2004. do 2005.godine.

3.8. PHP

PHP je prije značio (engl. *Personal Home Page*), a sada znači (engl. *PHP: Hypertext Preprocessor*). To je popularni skriptni jezik otvorenog koda koji je namijenjen za web razvoj. Razlikuje od JavaScripta koji je klijentski skriptni jezik, radi toga što se on izvršava na poslužitelju. PHP se koristi za čak 78% stranica čija je serverska strana poznata [8].

Začetnik PHP-a je Rasmus Lerdorf koji je 1994.godine napisao par CGI (Common Gateway Interface) programa u C-u, koji su mu služili kako bi održavao osobnu stranicu. Kasnije je proširio te programe da rade s web formama i da mogu komunicirati s bazom podataka i tako je PHP dobio svoje ime kao „Personal Home Interpreter“. Posljednja verzija PHP 8.1 je objavljena krajem 2021. godine.

Pomoću XAMPP programa koristiti se PHP bez da se mora instalirati na računalo, jer program simulira rad servera na računalu. PHP je kompatibilian s većinom servera, ali i s većinom baza podataka.

PHP može prikupljati podatke obrazaca, slati i primati kolačiće i generirati dinamički sadržaj stranice. Prilikom pisanja koda može se birati korištenje proceduralnog programiranja ili objektno orijentiranog programiranja ili čak kombinacije oba.

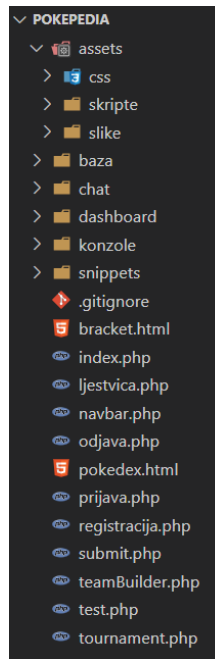
PHP kod se može pisati direktno u HTML datoteku ali se mora nalaziti između ovih oznaka `<?php ?>`, varijable se deklariraju pomoću `$` znaka. Primjer koda možemo vidjeti na slici 3.13.

<i>Linija</i>	<i>Kod</i>
1:	<code><html></code>
	<code><body></code>
	<code><?php</code>
	<code> \$zivotinje = array („Pas“, „Macka“, „Zec“, „Konj“);</code>
	<code> foreach(\$zivotinje as \$array_values){</code>
	<code> echo \$array_values . „
“; //ispis: Pas</code>
	<code> }</code>
2:	<code>?></code>
	<code></body></code>
3:	<code></head></code>

Sl. 3.8. PHP kod u HTML datoteci

4. IZRADA APLIKACIJE

U ovom poglavlju bit će opisan kod koji je korišten za izradu ovog web portala, korištene su različite tehnologije. Prvo što je napravljeno, kreiran je novi projekt pod nazivom Pokepedia. Struktura projekta je vidljiva na slici 4.1.



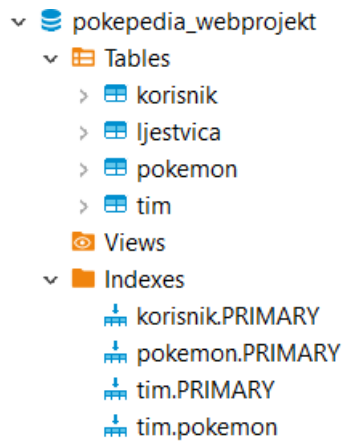
Sl. 4.1. Prikaz direktorija Pokepedia

4.1. Baza podataka

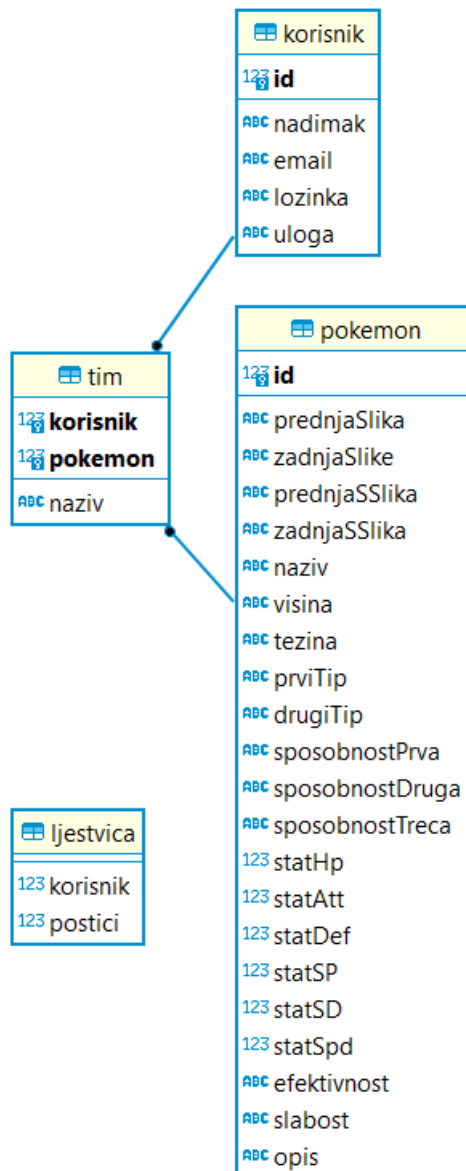
Program koji se koristio za kreiranje MySQL baze podataka je DBeaver. To je besplatni alat za korištenje u kojemu postoje brojne mogućnosti poput stvaranja, brisanja, ažuriranja te dohvaćanje podataka iz tablice. Program se koristi zajedno s „XAMPP“ programom kako bi se lokalno mogla kreirati baza, te kako bi se koristila i povezala s web aplikacijom.

Za ovaj rad kreirana je baza „pokepedia_webprojekt“ u kojoj se nalaze četiri tablice. Tablica „korisnik“ sadrži podatke o korisniku koji služe za prijavu, komunikaciju te za izgradnju tima, „ljestvica“ samo sadrži podatke o tome koji korisnik ima koliko bodova, „tim“ sadrži podatke koji pokemoni su vezani uz kojeg trenera, te tablica „pokemon“ koja u sebi sadrži 22 atributa vezana uz 151 pokemona iz prve generacije.

Na slici 4.2 može se vidjeti kako izgleda baza u programu DBeaver, vide se tablice koje kada se prošire sadrže svojstva, ključeve i ostale stvari, te vidjeti indexe tj. primarne ključeve. Na slici 4.3 može se vidjeti ER dijagram na kojemu su prikazane relacije između tablica.



Sl. 4.2. Prikaz baze u DBeaver izborniku



Sl. 4.3. ER dijagram baze podataka

Na slici 4.4 može se vidjeti kako je u DBeaveru napisan kod za stvaranje tablice „tim“, te kako se izmjenjuje tablica „korisnik“, koja ima strani ključ na tablicu „tim“.

```
create table tim (  
    id int(11) primary key auto_increment,  
    nazivTima varchar(40) not null,  
    prviPokemon int(11),  
    drugiPokemon int(11),  
    treciPokemon int(11),  
    foreign key(prviPokemon) references pokemon (id),  
    foreign key(drugiPokemon) references pokemon (id),  
    foreign key(treciPokemon) references pokemon (id)  
);  
  
alter table korisnik add column tim_id int(11);  
  
alter table korisnik add foreign key (tim_id) references tim(id);
```

Sl. 4.4. Prikaz MySQL koda

Na slici 4.5 može se vidjeti kako se u kodu spaja na bazu podataka. Ova datoteka se koristi u bilo kojoj datoteci koja se spaja na bazu, te mora biti unutar PHP koda.

```
<?php  
  
$hostname = "localhost";  
$username = "root";  
$password = "";  
$db = "pokepedia_webprojekt";  
  
$conn = new mysqli($hostname, $username, $password, $db);  
  
if($conn->connect_error) {  
    die ("Ne radi konekcija: " . $conn->connect_error);  
}
```

Sl. 4.5. Prikaz koda za spajanje na bazu

4.2. Početna stranica

Početna stranica se razlikuje za korisnike koji su prijavljeni i koji nisu, zajedničke stvari su im izgled stranice sa navbarom i footerom, te hamburger izbornik u gornjem desnom kutu. Za ne prijavljene korisnike hamburger izbornik sadrži samo prijavu i registraciju, dok za prijavljene korisnike sadrži stvari kao chat, team builder, turnir, ljestvica, odjava, te za neke korisnike i dashboard, isto tako prijavljenim korisnicima je između widgeta prikazan njihov tim, naravno ako ga imaju.

Na slici 4.6 može se vidjeti kako izgleda uspostavljanje konekcije i provjera postoji li tim odmah na početku. Slika 4.7 prikazuje provjeru je li korisnik prijavljen, te ako je provjera ima li tim. Dok slika 4.8 prikazuje kako se uključuju navbar i footer, tako se uključuju u svaku datoteku koja ih treba imati.

```
<?php
include_once './snippets/konekcija.php';

if (session_status() === PHP_SESSION_NONE) {
    session_start();
}

include './snippets/provjeraTima.php';

?>
```

Sl. 4.6. Otvaranje konekcije i provjera tima

```
<?php if (isset($_SESSION['loggedin']) && $_SESSION['loggedin'] == true) { ?>
<div class="row">
    <?php if ($_SESSION['imatim']) include './snippets/korisnik_team.php'; ?>
</div>
<?php } ?>
```

Sl. 4.7. Provjerava je li korisnik prijavljen i ima li tim

```
<header><?php include './navbar.php'; ?></header>
<footer class="container-fluid bg-dark text-center text-white text-lg-start mx-0px w-100">
    <?php include './snippets/footer.html'; ?>
</footer>
```

Sl. 4.8. Prikazuje uključenje navbara i footera

U navbaru se nalazi takozvani „hamburger“ izbornik, za ne prijavljenoga korisnika on samo sadrži prijavu i registraciju, dok za prijavljenog ima dosta više opcija koje su već prije navedene. Na slici 4.9 može se vidjeti kako on funkcionira, imamo provjeru je li korisnik ulogiran, ako je korisnik ulogiran možemo vidjeti koji izbornik će mu biti prikazan, a ako nije ulogiran bit će mu prikazan drugačiji izbornik.

```

<?php if (isset($_SESSION['loggedin']) && $_SESSION['loggedin'] == true) { ?>
<div class="navbar-nav ms-auto">
  <div class="nav-item dropdown dropstart" id="korisnikLogged">
    <a class="nav-link dropdown-toggle py-1" href="#" id="login_dropdown" role="button" data-bs-toggle="dropdown" aria-expanded="false">
      <?php echo $_SESSION['nadimak']; ?>
    </a>
    <div class="dropdown-menu dropdown-menu-dark" aria-labelledby="navbarDarkDropdownMenuLink">
      <a href="/chat/chat.php" class="dropdown-item" target="_blank">Chat</a>
      <a href="teambuilder.php" class="dropdown-item" target="_blank">Team Builder</a>
      <a href="tournament.php" class="dropdown-item">Tournament</a>
      <a href="ljestvica.php" class="dropdown-item">Ljestvica</a>
      <?php if ($_SESSION['role'] == 'admin' || $_SESSION['role'] == 'owner') { ?>
        <div class="dropdown-divider"></div>
        <a href="dashboard.php" class="dropdown-item" target="_blank">Dashboard</a>
        <div class="dropdown-divider"></div>
      <?php ?>
      <a href="odjava.php" class="dropdown-item">Odjava</a>
    </div>
  </div>
</div>
<?php ?>
if (!isset($_SESSION['loggedin'])) {
  echo '
  <div class="navbar-nav ms-auto">
    <div class="nav-item dropdown dropstart" id="korisnikNotLogged">
      <a class="nav-link dropdown-toggle py-1" href="#" id="login_dropdown" role="button" data-bs-toggle="dropdown" aria-expanded="false">
        <div class="dropdown-menu dropdown-menu-dark" aria-labelledby="navbarDarkDropdownMenuLink">
          <a class="dropdown-item" href="prijava.php">Prijava</a>
          <a class="dropdown-item" href="registracija.php">Registracija</a>
        </div>
      </div>
    </div>
  </div>';
} ?>

```

Sl. 4.9. Prikaz provjere u navbar-u

U footeru je uvijek uključena javascript koja služi kako bi se prikazao gumb koji vraća nazad na početak stranice. Na slici 4.10 može se vidi kako je uključeno u footer, a na slici 4.11 se može vidjeti Javascript kod za gumb koji nas vraća na početak stranice. Gumb za vraćanje nazad na početak stranice i Javascript datoteka su povezani preko „btn-back-to-top“ id-a.

```

<button type="button" class="btn btn-outline-dark btn-floating" id="btn-back-to-top">
  
</button>
<script src="./assets/skripte/pocetna.js"></script>

```

Sl. 4.10. Prikaz uključenog java script-a u footer

```

$(document).ready(function () {
  let mybutton = document.getElementById("btn-back-to-top");

  window.onscroll = function () {
    scrollFunction();
  };

  function scrollFunction() {
    if (document.body.scrollTop > 20 || document.documentElement.scrollTop > 20) {
      mybutton.style.display = "block";
    } else {
      mybutton.style.display = "none";
    }
  }

  mybutton.addEventListener("click", backToTop);

  function backToTop() {
    document.body.scrollTop = 0;
    document.documentElement.scrollTop = 0;
  }
});

```

Sl. 4.11. Prikaz java script koda za gumb

4.3. Pokédex

Nacionalni Pokedex na web stranici je dostupan svim korisnicima, nebitno jesu li oni prijavljeni ili ne. U njemu se nalaze svih 905 pokemona koji pripadaju osmoj generaciji, nova 9. generacija nije još dodana. Sve je napravljeno pomoću java scripta, te su svi podaci uzeti s „PokeApi“.

Na slici 4.12 može se vidjeti java script funkciju u kojoj pravimo listu pokemon kartica koje su klikabilne, te prikazuju sliku, id i naziv pokemona. Slika 4.13 prikazuje što se pokrene nakon što korisnik stisne na bilo koju pokemon kartu. Funkcija koja daje skočni prozor s prikaznom prednje i zadnje slike pokemona, prednje i zadnje slike „shiny“ varijante, te informacije kao prosječna visina, težina te tip pokemona, je vidljiva na slici 4.14.

```

const displayPokemon = (pokemon) => {
  console.log(pokemon);
  const pokemonHTMLString = pokemon
    .map(
      (pokeman) => `
        <li class="card" onclick="selectPokemon(${pokeman.id})">
          
          <h2 class="card-title">${pokeman.id}. ${pokeman.name}</h2>
        </li>
      `
    )
    .join("");
  pokedex.innerHTML = pokemonHTMLString;
};

```

Sl. 4.12. Funkcija za stvaranje pokemon kartica

```

const selectPokemon = async (id) => {
  const url = `https://pokeapi.co/api/v2/pokemon/${id}`;
  const res = await fetch(url);
  const pokeman = await res.json();
  displayPopup(pokeman);
};

```

Sl. 4.13. Funkcija za odabir pokemona

```

const displayPopup = (pokeman) => {
  console.log(pokeman);
  const type = pokeman.types.map((type) => type.type.name).join(", ");
  const image = pokeman.sprites["front_default"];
  const bimage = pokeman.sprites["back_default"];
  const simage = pokeman.sprites["front_shiny"];
  const sbimage = pokeman.sprites["back_shiny"];
  const htmlString = `
    <div class="popup">
      <button id="closeBtn" onclick="closePopup()">Close</button>
      <div class="card">
        <div class="column">
          
        </div>
        <div class="column">
          
        </div>
        <div class="column1">
          
        </div>
        <div class="column1">
          
        </div>
        <h2 class="card-title">${pokeman.id}. ${pokeman.name}</h2>
        <p><small>| Visina: </small> ${
          pokeman.height / 10
        } M | <small>Težina: </small> ${
          pokeman.weight / 10
        } Kg | <small>Tip: </small> ${type} | <small class="type ${type}"> </small> </p>
      </div>
    </div>
  `;
  pokedex.innerHTML = htmlString + pokedex.innerHTML;
};

```

Sl. 4.14. Funkcija koja prikazuje skočni prozor

4.4. Prijava i registracija

Prijava i registracija su potrebni za korisnika kako bi imao pristup svim funkcionalnostima web stranice. Registrirani korisnici mogu kreirati svoj tim od tri pokemona, a na izbor imaju 151 pokemona iz prve generacije. Isto tako imaju pristup chatu u kojemu mogu komunicirati sa svim prijavljenim korisnicima, mogu vidjeti ljestvicu s korisnicima i njihovim bodovima, te mogu sudjelovati u turniru od osam ljudi. Samo admini i jedan owner imaju pristup „dashboardu“ u kojemu mogu vidjeti sve informacije o korisnicima, admini ih mogu obrisati, a owner uz to ima i opciju da dodijeli ili makne status admina.

Na slici 4.15 može se vidjeti kako funkcioniraju provjere za registraciju korisnika. Prvo se provjerava podudaraju li se obje unese lozinke, ako je to istina ta lozinka se hashira, nakon toga se provjera jesu li uneseni nadimak i email već zauzeti u bazi, ako jesu neće biti prihvaćeni. Prije registracije korisnik mora prihvatiti uvjete korištenja, ako je sve to navede točno, podaci se upisuju u bazu, a korisnika se odvodi na drugi ekran kako bi se prijavio.

```
if ($pwd1 != $pwd2) {
    $error = "Lozinke se ne podudaraju!";
    $status = false;
} else {
    $hashed_password = password_hash($pwd1, PASSWORD_DEFAULT);

    $usernameTaken = "SELECT * FROM pokepedia_webprojekt.korisnik WHERE nadimak='$nadimak'";
    $doesExist = mysqli_query($conn, $usernameTaken);
    if (mysqli_num_rows($doesExist) > 0) {
        $error = "Nadimak je zauzet!";
        $status = false;
    }

    $userExists = "SELECT * FROM pokepedia_webprojekt.korisnik WHERE email='$email'";
    $doesExist = mysqli_query($conn, $userExists);
    if (mysqli_num_rows($doesExist) > 0) {
        $error = "Email je već zauzet!";
        $status = false;
    }

    if (isset($_POST['checkbox'])) {
        $error = "Morate prihvatiti uvjete za registraciju!";
        $status = false;
    }

    if ($status) {
        $query = "INSERT INTO pokepedia_webprojekt.korisnik (nadimak, email, lozinka, uloga) VALUES ('$nadimak', '$email', '$hashed_password', 'user')";
        $results = mysqli_query($conn, $query);
        if (!$results) {
            printf("Error message: %s\n", $conn->error);
        }

        header('Location: ./prijava.php');
    }
}
```

Sl. 4.15. Prikaz koda za registracijske provjere

Slika 4.16 prikazuje kod za prijavu korisnika. Na početku se provjera jesu li email i lozinka uopće uneseni, ako jesu s while petljom provjeravaju informacije unesenih podataka te ako su dobri user se ulogira na stranicu, nakon toga se provjerava ima li korisnik tim, i nebitno o tome ima li ili nema korisnika se vodi na početnu stranicu tj. na index. Na slici 4.17 može se vidjeti kako se spremaju kolačići za email i lozinku, jedino ako je korisnik odabrao pri prijavi da ga sustav zapamti.

```
if (isset($_POST['login']) && !empty($_POST['email']) && !empty($_POST['lozinka'])) {
    $sql = "SELECT * FROM pokepedia_webprojekt.korisnik";
    $q = mysqli_query($conn, $sql);

    while ($line = mysqli_fetch_array($q, MYSQLI_ASSOC)) {
        if ($_POST['email'] == $line['email'] && password_verify($_POST['lozinka'], $line['lozinka'])) {
            $_SESSION['loggedin'] = true;
            $_SESSION['email'] = $_POST['email'];
            $_SESSION['nadimak'] = $line['nadimak'];
            $_SESSION['role'] = $line['uloga'];
            $_SESSION['id'] = $line['id'];

            include './snippets/provjeraTima.php';

            header('Location: ./index.php');
        } else {
            $msg = 'Wrong login details.';
        }
    }
}
```

Sl. 4.16. Prikazuje se postupak prijave

```
if (!empty($_POST["remember"])) {
    setcookie("email", $_POST["email"], time() + 3600);
    setcookie("lozinka", $_POST["lozinka"], time() + 3600);
} else {
    setcookie("email", "");
    setcookie("lozinka", "");
}
```

Sl. 4.17. Prikazuje se kako se spremaju kolačići

4.5. Izgradnja tima

Svim prijavljenim korisnicima nudi se opcija izgradnja vlastitog tima od tri pokemona. Korisnici svoj tim mogu napraviti samo od 151 pokemona iz prve generacije, radi toga što se koristi ručno napisana lokalna baza kako bi se korisniku pružilo puno više informacija nego PokeApi. Neki od svojstava koji nisu bili dostupni na api su statusi, efektivnost, slabost, visina, težina, te opis pokemona.

Na slici 4.18 može se vidjeti funkciju koja provjera koji je pokemon u timu, a koji nije, kako bi korisnik znao što može dodati u svoj tim od tri pokemona, a slika 4.19 prikazuje kako se ta funkcija u kodu koristi kako bi znali koji gumb prikazati korisniku. Slika 4.20 prikazuje kako funkcionira pretraživanje, pretraživati se može po nazivu, tipu te id-u pokemona. Kako je u bazu dodano četiri slike, dvije za normalnog pokemona i dvije za shiny pokemona, te kako bi obje bile vidljive korisniku bez klikanja, stavljeno je da su dvije prednje slike vidljive stalno, a na ulazak miša na sliku pokaže se zadnja slika pokemona, kako četiri slike ne bi zauzimale previše mjesta. To može se vidjeti na slici 4.21.

```
function isInTeam($pokeId, $team)
{
    $isInTeam = false;
    foreach ($team as &$amp;teamPoke) {
        if ($pokeId == $teamPoke) {
            $isInTeam = true;
        }
    }
    return $isInTeam;
}
```

Sl. 4.18. Prikaz funkcije za provjeru je li pokemon u timu

```
<?php
if (isInTeam($row['id'], $pokemoniutimu)) { ?>
    <form action="./snippets/ukloni.php?id=?php echo $row['id']; ?>" method="POST">
        <button class="btn btn-danger" type="submit">Ukloni</button>
    </form>
<?php } else { ?>
    <form action="./snippets/dodaj.php?id=?php echo $row['id']; ?>" method="POST">
        <button class="btn btn-dark" <?php if ($timpun) echo 'disabled'; ?> type="submit">Dodaj</button>
    </form>
<?php } ?>
```

Sl. 4.19. Korištenje provjere za postavljanje gumbova

```

if (isset($_POST['search'])) {

    $search_term = mysqli_real_escape_string($conn, $_POST['search_box']);

    if ($search_term == "") $sql;
    else {
        $sql .= " WHERE naziv='{$search_term}'";
        $sql .= " OR id='{$search_term}'";
        $sql .= " OR prviTip='{$search_term}'";
        $sql .= " OR drugiTip='{$search_term}'";
    }
}
}

```

Sl. 4.20. Prikaz funkcije za pretraživanje pokemona

```

<script>
$(document).ready(function() {
    $("#prednjaS-<?php echo $row['id']; ?>").on("mouseenter", function() {
        $("#prednjaS-<?php echo $row['id']; ?>").removeClass("d-inline").addClass("d-none");
        $("#zadnjaS-<?php echo $row['id']; ?>").removeClass("d-none").addClass("d-inline");
    }).on("mouseleave", function() {
        $("#prednjaS-<?php echo $row['id']; ?>").removeClass("d-none").addClass("d-inline");
        $("#zadnjaS-<?php echo $row['id']; ?>").addClass("d-none").removeClass("d-inline");
    });

    $("#prednjaSS-<?php echo $row['id']; ?>").on("mouseenter", function() {
        $("#prednjaSS-<?php echo $row['id']; ?>").removeClass("d-inline").addClass("d-none");
        $("#zadnjaSS-<?php echo $row['id']; ?>").removeClass("d-none").addClass("d-inline");
    }).on("mouseleave", function() {
        $("#prednjaSS-<?php echo $row['id']; ?>").removeClass("d-none").addClass("d-inline");
        $("#zadnjaSS-<?php echo $row['id']; ?>").addClass("d-none").removeClass("d-inline");
    });
});
</script>

```

Sl. 4.21. Prikaz kako se na ulaz miša promijeni slika

4.6. Chat

Kao i izgradnja tima, chat je dostupan samo prijavljenim korisnicima. U njemu korisnici mogu komunicirati sa svim ostalim prijavljenim korisnicima.

Na slici 4.22 može se vidjeti java script kod u kojemu se pritisak gumba aktivira slanje poruke, tj. poziva se post.php koji služi za dodavanje poruke u log.html, to se može vidjeti na slici 4.23. Ispod toga imamo ajax kod koji konstantno bez ažuriranja stranice povlači novo dodan tekst u log.html i prikazuje ga korisnicima. Zadnje na kraju postoji opcija za napuštanje chata, i tu se poziva logoutChat.php koji u log.html dodaje tekst da je korisnik izašao, preusmjeri korisnika na početnu stranicu, prikazano na slici 4.24.

```
<script type="text/javascript">
$(document).ready(function() {
    $("#submitmsg").click(function() {
        var clientmsg = $("#usermsg").val();
        $.post("post.php", {
            text: clientmsg
        });
        $("#usermsg").val("");
        return false;
    });

    function loadLog() {

        $.ajax({
            url: "./log.html",
            cache: false,
            success: function(html) {
                $("#chatbox").html(html);
            }
        });

        setInterval(loadLog);
    }

    $("#exit").click(function() {
        var exit = confirm("Jeste li sigurni da želite napustiti chat?");
        if (exit == true) window.location = "./logoutChat.php";
    });
});
</script>
```

Sl. 4.22. JavaScript kod za chat

```
<?php
session_start();
if(isset($_SESSION['nadimak'])){
    $text = $_POST['text'];

    $text_message = "<div class='msgIn'><span class='chat-time'>".date("g:i A")."</span> <b class='user-name'>".$_SESSION['nadimak']
    file_put_contents("log.html", $text_message, FILE_APPEND | LOCK_EX);
}
?>
```

Sl. 4.23. Post.php kod

```
<?php
if (session_status() === PHP_SESSION_NONE) {
    session_start();
}

$loglogout_message = "<div class='msgIn'><span class='left-info'>Korisnik: <b class='user-name-left'>".$_SESSION['nadimak']."</b>
file_put_contents("./log.html", $loglogout_message, FILE_APPEND | LOCK_EX);

header('Location: ../index.php');
```

Sl. 4.24. LogoutChat.php kod

4.7. Upravljača ploča

Upravljačkoj ploči ili dashboardu pristup nemaju ni svi prijavljeni korisnici, nego samo admini i jedan vlasnik („owner“). Tamo su prikazane osnovne informacije kao email, nadimak, uloga te na click i tim koji je vezan uz tog korisnika. Admini imaju mogućnost brisanja korisnika, dok vlasnik uz brisanje ima i mogućnost postaviti korisnike za admine i maknuti ih s te pozicije.

Na slici 4.25 može se vidjeti kako se provjera je li prijavljeni korisnik admin ili vlasnik, te što će njemu biti ponuđeno od opcija. Slika 4.26 prikazuje kako se korisnik briše iz sustava, vidi se da se prvo mora obrisati njegov tim ako postoji, radi stranih ključeva.

```
<?php if ($korisnikRow['uloga'] != 'owner') { ?>
<div class="row pt-3 mt-3">
  <?php if ($_SESSION['role'] == 'owner') {
    echo '<div class="col-md-6">';
  } else {
    echo '<div class="col-md-12">';
  } ?>
  <a href="./obrisiKorisnika.php?id=<?php echo $korisnikRow['id']; ?>", id="obrisiKorisnika-<?php echo $korisnikRow['id'];
</div>
<div class=" col-md-6">
  <?php if ($_SESSION['role'] == 'owner') {
    if ($korisnikRow['uloga'] != 'admin') { ?>
      <a href="./postaviAdmina.php?id=<?php echo $korisnikRow['id']; ?>&isNotAdmin=true", id="postaviAdmina" class="btn
      <?php } else { ?>
      <a href="./postaviAdmina.php?id=<?php echo $korisnikRow['id']; ?>&isNotAdmin=false", id="ukloniAdmina" class="btn
      <?php }
    } ?>
  } ?>
</div>
</div>
<?php } ?>
```

Sl. 4.25. Provjera za admina i vlasnika

```
if (isset($_SESSION['id'])) {
  $provjeraTim = "SELECT * FROM tim WHERE korisnik='$_SESSION[id]'";
  $result = mysqli_query($conn, $provjeraTim);
  if (mysqli_num_rows($result) > 0) {
    $delTimQuery = "DELETE FROM tim WHERE korisnik='$userId' AND pokemon BETWEEN 1 AND 151";
    if (mysqli_query($conn, $delTimQuery)) {
      header('Location: ./dashboard.php');
    } else {
      echo "Error: " . $delTimQuery . ":-" . mysqli_error($conn);
    }
  }
  $korisnikDel = "DELETE FROM korisnik WHERE id='$userId'";
  if (mysqli_query($conn, $korisnikDel)) {
    header('Location: ./dashboard.php');
  } else {
    echo "Error: " . $query . ":-" . mysqli_error($conn);
  }
}
```

Sl. 4.26. Brisanje korisnika

Kako se radi postavljanje admina je vidljivo na slici 4.27 . Postoji još jedna stvar koju mogu vidjeti admin i vlasnik, a to je da klikom na korisnikov nadimak sakrivaju se trenutni podaci i pokaže korisnikov tim, ako postoji, ta promjena je ostvorena u java scriptu i prikazano je na slici 4.28.

```
<?php
include './konekcija.php';

session_start();
$userId = $_GET['id'];
$checkAdmin = $_GET['isNotAdmin'];

$strSQL = $checkAdmin == "true" ? "UPDATE korisnik SET uloga='admin' WHERE id='$userId'" : $strSQL = "UPDATE korisnik SET uloga='ko

if (mysqli_query($conn, $strSQL)) {
    header('Location: ./dashboard.php');
} else {
    echo "Error: " . $strSQL . ":-" . mysqli_error($conn);
}
}
```

Sl. 4.27. Postavljanje admina

```
<script>
$( ".ulogaa-<?php echo $korisnikRow['id']; ?>" ).on("click", function() {
    $("#korisnikCardPrednja-<?php echo $korisnikRow['id']; ?>").addClass("d-none");
    $("#korisnikCardZadnja-<?php echo $korisnikRow['id']; ?>").removeClass("d-none");
});
$("#korisnikCardZadnja-<?php echo $korisnikRow['id']; ?>").on("click", function() {
    $(this).addClass("d-none");
    $("#korisnikCardPrednja-<?php echo $korisnikRow['id']; ?>").removeClass("d-none");
});
</script>
```

Sl. 4.28. Promjena prikazanih podataka

4.8. Turnir i ljestvica

Ljestvica i turnir su dvije nove funkcije na stranici koje su povezane, povezane su tako da kada se završi turnir igrači i bodovi se dodaju u ljestvicu kako bi ostali prijavljeni korisnici mogli vidjeti tko ima koliko bodova. Igrači su poredani tako da je na vrhu igrač s najviše bodova, pa se bodovi prema dolje smanjuju, na ljestvici su prikazani samo igrači koji su ostvarili bodove u turniru. Turnir se sastoji od osam igrača, gdje na kraju pobjednik dobiva 3 boda, a drugoplasirani 1.5 bod. Pri odabiru igrača koji će biti u turniru vide se nazivi pokemona koje koriste u timu.

Stranica turnira se sastoji od dva dijela. Prvi dio je tablica u kojoj se nalaze svi korisnici koji su registrirani, njihov pokemon tim, ako postoji, te gumb s kojim se oni dodaju u turnir. Prikaz tablice se može vidjeti na slici 4.29.

```

<table class="table table-hover">
  <tr>
    <th>Nadimak</th>
    <th>Tim</th>
    <th>Akcija</th>
  </tr>
  <?php
  foreach ($korisnik as $key => $value) {
    $pokes = [];

    foreach ($tim as $key2 => $value2) {
      if ($key == $value2[0]) {
        array_push($pokes, $value2[1]);
      }
    }

    if(count($pokes) == 0) {
      array_push($pokes, "Nema pokemon");
    }
    $username = $value['nadimak'];

    echo "<tr>";
    echo "<td>" . $value['nadimak'] . "</td>";
    echo "<td>" . implode(", ", $pokes) . "</td>";
    echo "<td>" .
    "<button onclick=pickUser($key, '$username') class='btn btn-primary' >Odaberi</button>";
    <input type='hidden' value='' . $key . "' name='id'>
    <input type='hidden' value='' . $value['nadimak'] . "' name='nadimak'>
    </td>";
    echo "</tr>";
  }
  ?>
</table>

```

Sl. 4.29. Kod za tablicu u turniru

Ispod tablice u turniru se nalaze dva gumba „Reset“ i „Start“, oni služe za ponovno postavljanje turnira ili pokretanje istoga. Ispod toga se nalazi prostor u kojemu je struktura turnira gdje se osam igrača dodaje, te kada se osam mjesta popuni turnir se može pokrenuti. Na slici 4.30 može se vidjeti kako izgleda kod za ta dva gumba i gdje se uključuje bracket.html koji predstavlja izgled turnira, dok slika 4.31 pokazuje isječak koda iz bracket.html gdje vidimo kako se sve zapravo „crtalo“ u redovima (<tr>) i kako su linije između čelija nacrtani s obrubima drugih čelija (<td>).

```

<div class="column" id="div-game-control" style="margin-top: 20px;">
  <div class="game">
    <div class="control" id="control">
      <button class="btn btn-danger" onclick=resetGame()>RESET</button>
      <button class="btn btn-success" id="btn-start-stop" onclick=startGame()>START</button>
    </div>
    <div class="play-area" id="play-area">
    </div>
    <div class="play">
      <?php include('./bracket.html');?>
    </div>
    <div id="win-area"></div>
  </div>
</div>

```

Sl. 4.30. Gumbi i uključenje izgleda turnira

```

<tr id="tr-7">
  <td></td>
  <td></td>
  <td></td>
  <td></td>
  <td></td>
  <td></td>
  <td class="b-r"></td>
  <td class="b-b"></td>
  <td class="field" colspan="2" rowspan="2" id="f-l">
    <button class="btn-player btn btn-info" disabled></button>
  </td>
  <td class="b-b"></td>
  <td class="b-b"></td>
  <td class="field" colspan="2" rowspan="2" id="w">
    <button class="btn-player btn btn-info" disabled></button>
  </td>
  <td class="b-b"></td>
  <td class="b-b"></td>
  <td class="field" colspan="2" rowspan="2" id="f-r">
    <button class="btn-player btn btn-info" disabled></button>
  </td>
  <td class="b-b"></td>
  <td class="b-l"></td>
  <td></td>
  <td></td>
  <td></td>
  <td></td>
  <td></td>
  <td></td>
  <td></td>
</tr>

```

Sl. 4.31. Isječak iz bracket.html

Gumbovi „Start“ i „Reset“ svaki pozivaju jednu java script funkciju. Java script za početak igre prvo makne „disabled“ atribut sa svim ćelijama u turniru kako bi bile klikabilne, nakon toga sakrije tablicu i samog sebe kako bi turnir i gumb „Reset“ bili jedini na ekranu, to se može vidjeti na slici 4.32. Java script za „Reset“ button je jednostavna funkcija koja samo sve vrati na početak i ponovno učita stranicu, ona je vidljiva na slici 4.33.

```

function startGame(){
  var cell;

  for(var i = 0; i < cellAdd.length; i++){
    cell = document.getElementById(cellAdd[i]).firstChild;
    cell.removeAttribute("disabled");
  }

  var playersTableDiv = document.getElementById('div-player-table');
  playersTableDiv.style.display = "none";

  btnStart.style.display = 'none';
}

```

Sl. 4.32. startGame funkcija


```

resetGame = function(){
    playersSelected = [];
    winCount = 0;
    score = [0, 0];
    userswon = [];

    location.reload();
}

```

Sl. 4.33. resetGame funkcija

Gumb „Odaberi“ poziva određenu java script funkciju koja dodaje igrača u turnir, ako je igrač već dodan neće moći opet biti dodan, dodaje se novi gumb koji se nalazi u odgovarajućoj ćeliji koja prikazuje korisničko ime. Gumb sa korisničkim imenom je onemogućen i aktivira se tek kada igra počne, tj. kada korisnik klikne na gumb „Start“. Funkcija za odabiranje korisnika za turnir se može vidjeti na slici 4.34. Pritiskom na gumb s korisničkim imenom poziva se još jedna java script funkcija koja služi za pomicanje igrača koji su pobijedili u toj rundi. Funkcija deklarira dvije varijable „oldCell“ i „winCell“ koje predstavljaju prethodnu i pobjedničku ćeliju, nakon toga ulazi u petlju u kojoj dohvaća prošle ćelije preko id-a, te se iskorištene ćelije onemogućuju i boja ih u crveno. Kao i u prethodnoj funkciji kreira se novi gumb koji se sada nalazi u novoj ćeliji koja sadrži nadimak igrača koji je prošao u drugi krug, ako sljedeća ćelija nije posljednja lista se ažurira, a ako je ćelija više nije klikabilna, boja se mijenja u zelenu te se poziva winGame funkcija, to je sve vidljivo na slici 4.35. WinGame funkcija kreira novi „Submit“ gumb, predaje mu korisnike koji su osvojili prvo i treće mjesto te na klik poziva submit.php datoteku, slika 4.36 to prikazuje.

Submit.php datoteka provjera jesu li joj dobro predana dva igrača, ako jesu nastavlja se dalje tako da se dohvaća sve iz tablice ljestvica i sprema se u jedan niz, ispod toga se nalazi petlja koja prolazi kroz dobiven rezultat te u niz sprema korisnika i bodove, korisnika kao ključ. Kreiraju se dvije varijable s vrijednostima 3 i 1.5, nakon toga prolazi se kroz niz dok dobiveni igrači ne odgovaraju igračima u nizu, te se onda njima dodaju bodovi za osvojeno prvo ili treće mjesto i varijable se postavljaju na nulu. Zadnja provjera je ako varijable nisu postavljene na nulu znači da igrač nije pronađen, tj. do sada nije ostvario bodove na turniru, te ga se onda ubacuje u tablicu. Na kraju smo preusmjereni na ljestvica.php. Isječak koda u kojemu se rade provjere se može vidjeti na slici 4.37.

```

function pickUser(userId, username){
    for(var i = 0; i < playersSelected.length; i++){
        if(playersSelected[i] == userId){
            return;
        }
    }
    playersSelected.push(userId);

    var btnPlayer = "<button class=\"btn-player btn btn-primary\" disabled onclick=\"moveCell(" + userId + " +
    var cell = document.getElementById(cellAdd[playersSelected.length-1]);
    cell.innerHTML = btnPlayer;
}

```

Sl. 4.34. PickUser funkcija

```

function moveCell(userId, username, nextCell){
    var oldCell;
    var winCell;
    for(i=0; i<oldCell_lst[nextCell].length; i++){
        oldCell = document.getElementById(oldCell_lst[nextCell][i]).firstChild;
        oldCell.disabled = true;
        oldCell.setAttribute("class", "btn-player btn btn-danger");
    }

    var btnPlayer = "<button class=\"btn-player btn btn-primary\" onclick=\"moveCell(" + userId + ", ' " +
    var cell = document.getElementById(nextCell);
    cell.innerHTML = btnPlayer;

    if(nextCell != 'w'){
        usersWon[1] = usersWon[0];
        usersWon[0] = userId;
    }

    if(nextCell == 'w'){
        winCell = document.getElementById('w').firstChild;
        winCell.disabled = true;
        winCell.setAttribute("style", "background-color: green;");

        winGame();
    }
}

```

Sl. 4.35. MoveCell funkcija

```

winGame = function(){
    var submitBtn = document.createElement("a");
    submitBtn.setAttribute("class", "btn btn-success");
    submitBtn.setAttribute("href", "submit.php?user1=" + usersWon[0] + "&user2=" + usersWon[1]);
    submitBtn.innerHTML = "Submit";

    controlDiv.appendChild(submitBtn);
}

```

Sl. 4.36. WinGame funkcija

```

$score1 = 3;
$score2 = 1.5;

foreach($ljestvica as $key => $value){
    if($key == $_GET['user1']){
        $score1 += $value;

        $update_ljestvica = "UPDATE ljestvica SET postici = $score1 WHERE korisnik = $key";
        $conn->query($update_ljestvica);

        $score1 = 0;
    }
    if($key == $_GET['user2']){
        $score2 += $value;

        $update_ljestvica = "UPDATE ljestvica SET postici = $score2 WHERE korisnik = $key";
        $conn->query($update_ljestvica);

        $score2 = 0;
    }
}

if($score1 != 0){
    $insert_ljestvica = "INSERT INTO ljestvica (korisnik, postici) VALUES (" . $_GET['user1'] . ", $score1)";
    $conn->query($insert_ljestvica);
}

if($score2 != 0){
    $insert_ljestvica = "INSERT INTO ljestvica (korisnik, postici) VALUES (" . $_GET['user2'] . ", $score2)";
    $conn->query($insert_ljestvica);
}

```

Sl. 4.37. Provjere u submit.php

Za razliku od turnira ljestvica je vrlo jednostavna. Na slici 4.38 može se vidjeti kako se podaci iz baze dohvaćaju i spremaju u novi niz. Dohvaća se baza korisnik, sprema se u varijablu i stvara se prazan niz korisnik. U while petlji se prolazi kroz dohvaćene podatke i onda se parametar id predaje kao ključ, a nadimak kao vrijednost toga ključa. Nakon toga kroz petlju iz tablice ljestvica svakom korisniku se pridružuju Bodovi koje je postigao. Kako bi ljestvica bila poredana tako da su na vrhu igrači s najviše bodova koristi se „arsort“. Slika 4.39 prikazuje kako je kreirana ljestvica, te gdje su prosljeđeni „skriveni“ podaci sa slike 4.29.

```

$get_korisnik = "SELECT * FROM korisnik";
$result_korisnik = $conn->query($get_korisnik);
$korisnik = [];

while ($row = $result_korisnik->fetch_assoc()) {
    $korisnik[$row['id']] = $row['nadimak'];
}

$get_ljestvica = "SELECT * FROM ljestvica";
$result_ljestvica = $conn->query($get_ljestvica);
$ljestvica = [];

while ($row = $result_ljestvica->fetch_assoc()) {
    $ljestvica[$korisnik[$row['korisnik']]] = $row['postici'];
}

arsort($ljestvica);

```

Sl. 4.38. Dohvaćanje podataka za ljestvicu

```
<table class="table table-striped">
  <tr>
    <th>Nadimak</th>
    <th>Bodovi</th>
  </tr>
  <?php
    foreach ($ljestvica as $key => $value) {
      echo "<tr>";
      echo "<td>$key</td>";
      echo "<td>$value</td>";
      echo "</tr>";
    }
  ?>
</table>
```

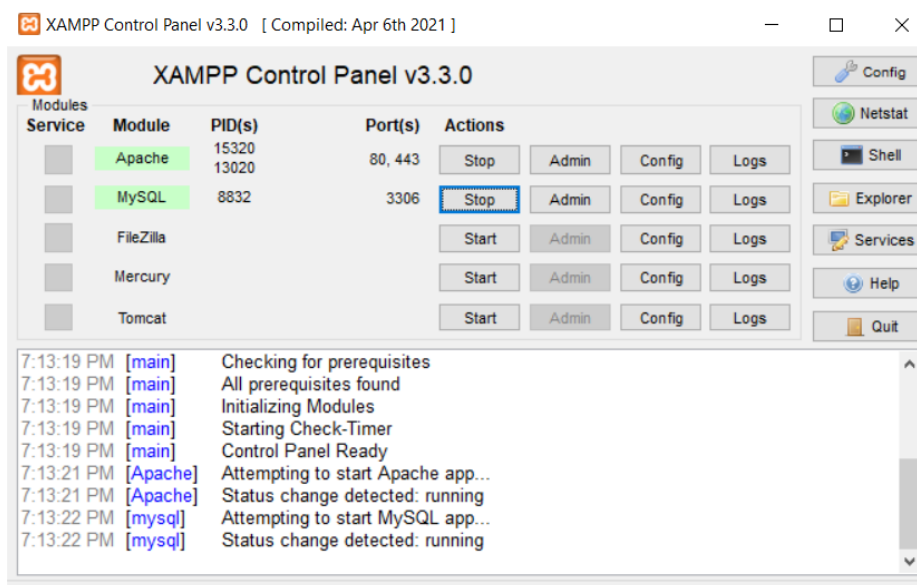
Sl. 4.39. Prikaz ljestvice

5. KORIŠTENJE APLIKACIJE

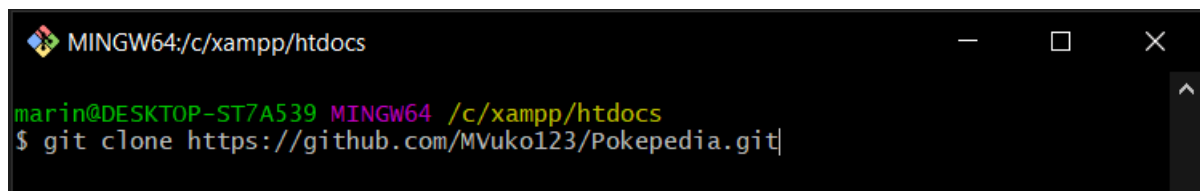
Prva stvar koja se mora napraviti kako bi se web aplikacija pokrenula je skinuti i instalirati XAMPP razvojno okruženje. To služi kako bi mogli simulirati rad servera, preko xampp controla potrebno je uključiti MySQL modul na kojemu se nalazi naša baza podataka i Apache web server kako bi mogli lokalno pokrenuti, kako XAMPP treba izgledati vidi se na slici 5.1.

Web aplikaciju je potrebno skinuti sa githuba, te ju smjestiti u htdocs datoteku koja pripada XAMPP-u. Najlakši način za to je otvoriti git bash u htdocsu te upisati samo jednu naredbu da se cijeli projekt postavi u htdocs, to je vidljivo na slici 5.2.

Nakon što se oba koraka obave, potrebno je učitati sql bazu koja je preuzeta zajedno s projektom, i u projektu se nalazi u direktoriju baza pod nazivnom „pokepedia_webprojekt.sql“. Baza se može uvesti preko DBeavera, ali ako korisnik ne posjeduje DBeaver moguće je nakon pokretanja XAMPP-a u preglednik upisati „localhost/phpmyadmin“, te tu pronaći opciju „import“.



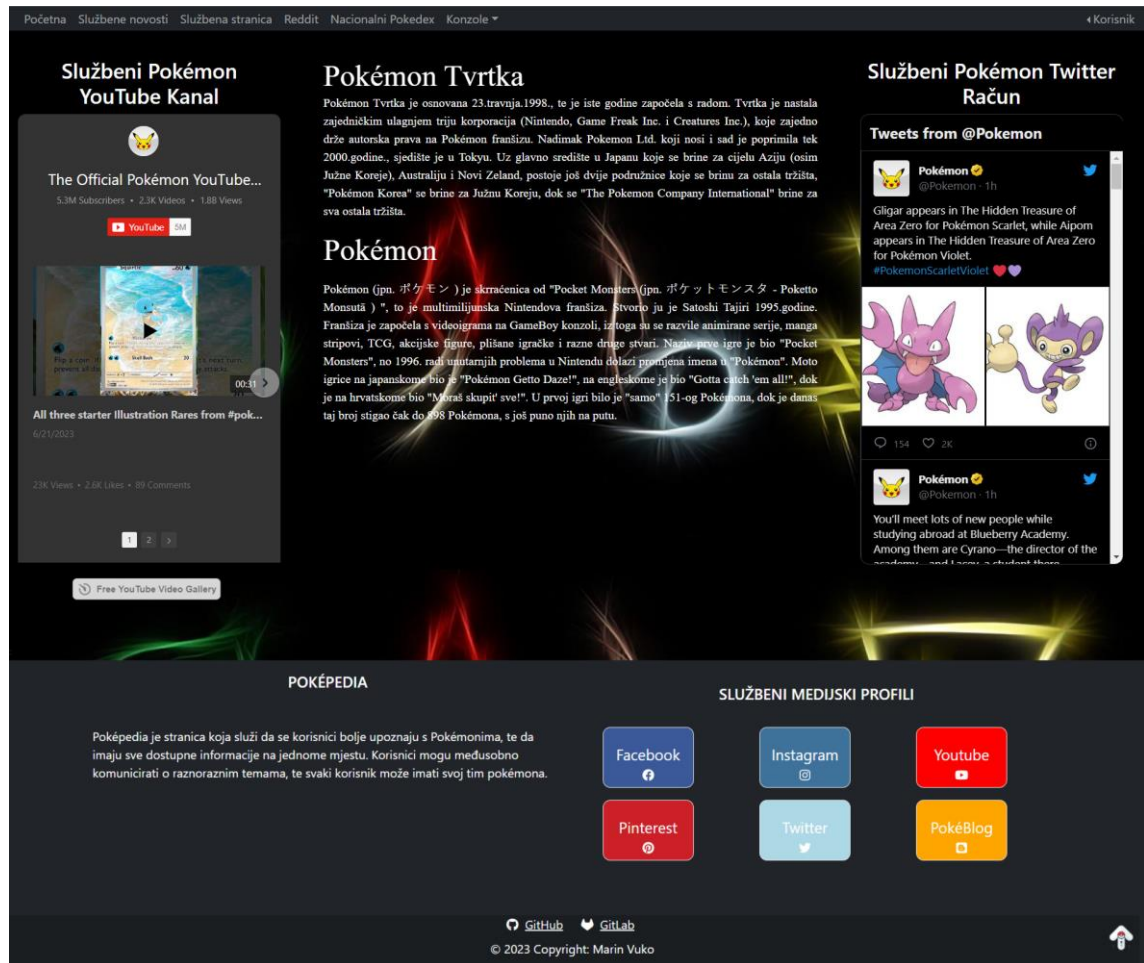
Sl. 5.1. Izgled XAMPP sučelja



Sl. 5.2. Git naredba za kloniranje repozitorija

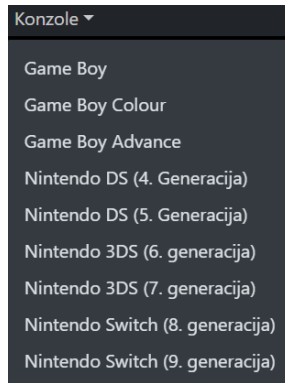
5.1. Neregistrirani korisnik

Neregistrirani korisnik će na stranici imati ograničeno što može raditi u usporedbi sa registriranim korisnicima. Izgled početne stranice je na slici 5.3 i jako je sličan kao i prijavljenim korisnicima, ono što se odmah može primijetiti da fali je prikaz tima između widgeta, te u gornjem desnom kutu stoji korisnik, a ne korisničko ime iza kojega su sve mogućnosti. Pozadinska slika se ne ponavlja, nego su dvije slike morale biti spojene u jednu kako bi se cijela stranica vidjela.

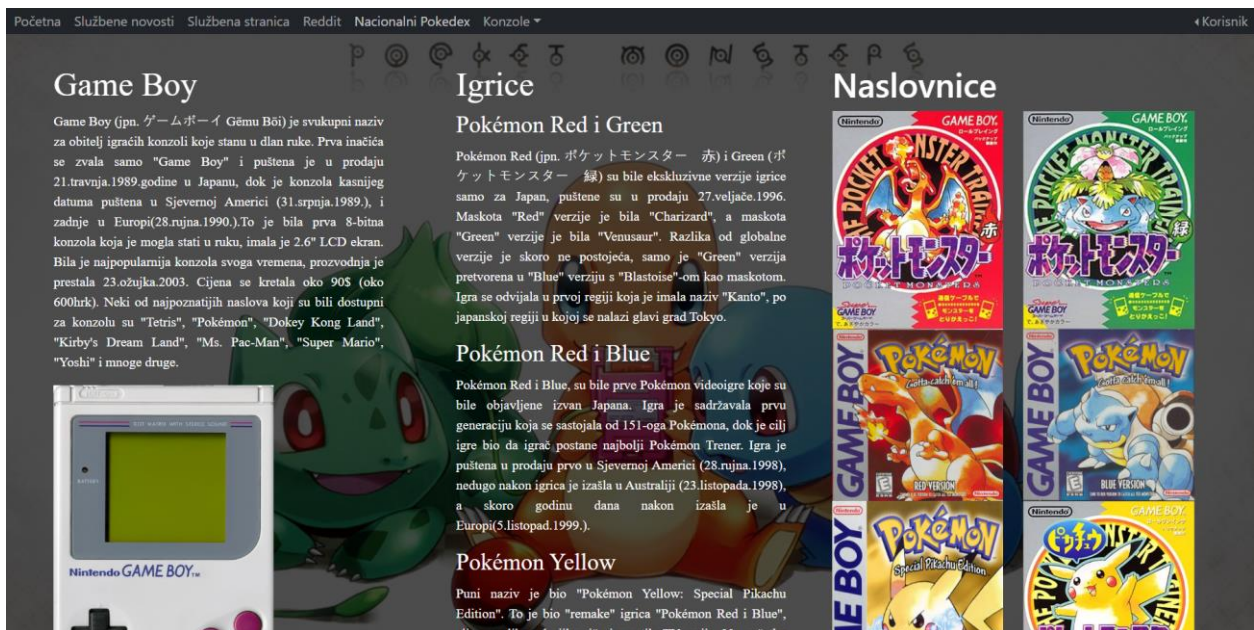


Sl. 5.3. Izgled početne stranice za neregistrinog korisnika

Ne registrirani korisnik može imati interakciju s dva widgeta, kliknuti na footeru i navbaru vanjske linkove koji na službene stranice te reddit. Korisnik može otvori padajući izbornik za konzole te pročitati o svakoj igri i generaciji pokemona, te o konzoli na kojoj su te igre bile puštene. Izbornik je vidljiv na slici 5.4, a jedna od stranica na slici 5.5. Klikom na početna u navbaru može se vratiti na početnu stranicu, ili navigirati na ostale stvari na stranici.











Sl. 5.4. Konzole drop-down menu



Sl. 5.5. Stranica o prvoj generaciji pokemona

Uz čitanje o igricama i konzolama, neregistriranom korisniku je dostupan „Nacionalni Pokédex“, kako izgleda kada se otvori je vidljivo na slici 5.6. Tu su prikazani svi pokemona do kraja osme generacije, znači 905 pokemona. Pokemona su prikazani uz pomoć api-a, a klikom na bilo kojeg pokemona otvara se skočni prozor na kojemu je više informacija o pokemonu, može se vidjeti na slici 5.7. Klikom na naslov „Pokédex“ može se vratiti na početnu stranicu.

POKÉDEX			
 1. Bulbasaur	 2. Ivysaur	 3. Venusaur	 4. Charmander
 5. Charmeleon	 6. Charizard	 7. Squirtle	 8. Wartortle

Sl. 5.6. Izgled nacionalnog pokedexa



Sl. 5.7. Skočni prozor o pokemonu

Klikom na „Korisnik“ u gornjem desnom kutu, otvara se mali hamburger izbornik koji sadrži dvije opcije, prijavu i registraciju. Ako je korisnik nov na stranici odabrat će registraciju, mora unijeti sve tražene podatke te prihvatiti uvjete korištenja, ako korisnik već posjeduje račun može koristiti link na dnu, sve je to vidljivo na slici 5.8.

REGISTRACIJA
Unesite vaše podatke

Vaš nadimak

Unesite Vaš Email

Unesite Vašu Lozinka

Ponovite Vašu Lozinka

Prihvaćam sve navedene [uvjete korištenja.](#)

Registriraj se

Posjeduješ račun? [Prijavi se](#)

Sl. 5.8. Izgled forme za registraciju

Ako korisnik već posjeduje račun može odabrati prijavu, isto tako nakon registracije korisnik će biti odveden na stranicu za prijavu kako bi dobio pristup svim funkcionalnostima na stranici. Forma za prijavu je vidljiva na slici 5.9. Dosta je manja nego ona za registraciju, jer su potrebna samo dva podatka, ako korisnik želi biti zapamćen može označiti kvadratić kraj zapamti me.

PRIJAVA
Unesite vaš email i lozinku!

Email

Lozinka

Zapamti me.

Prijava se

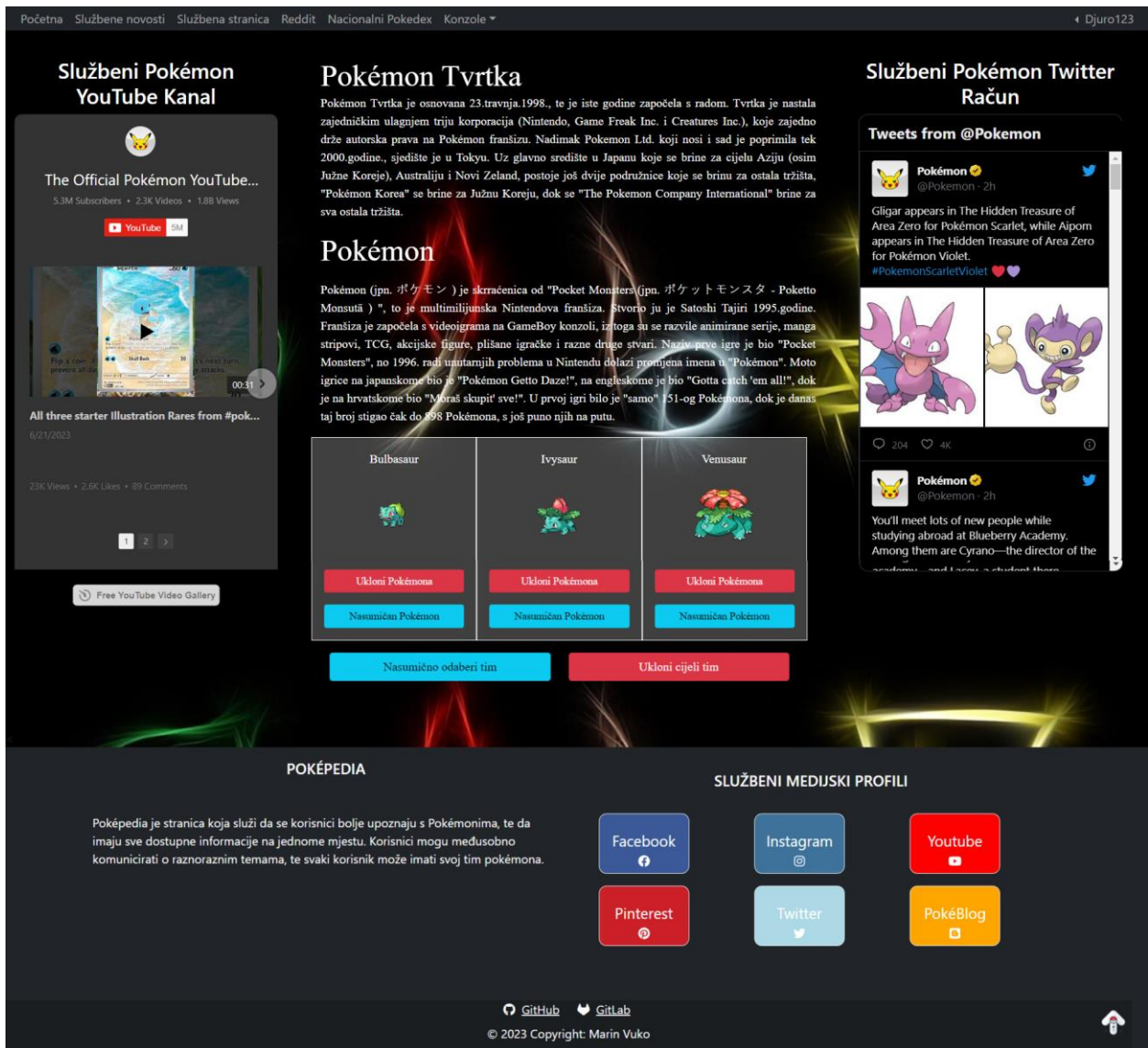
Ne posjeduješ račun? [Registriraj se](#)

Sl. 5.9. Izgled forme za prijavu

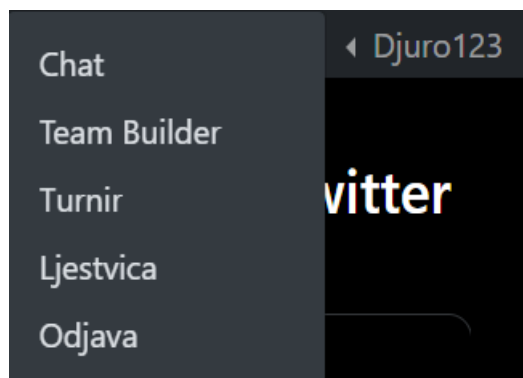
5.2. Registrirani korisnik

Nakon što je izvršena registracija i prijava na stranicu, korisnik dobiva nove stvari na stranici koje može koristiti. Prva razlika koja se vidi čim se otvori stranice je ta da na početnoj stranici između widgeta se sada nalazi korisnikom tim, naravno ako ga ima, te u gornjem desnom kutu sada stoji njegov nadimak na koji kada se klikne otvori izbornik s opcijama za chat, team builder, turnir, ljestvica i odjava. Izgled takve početne stranice je vidljiv na slici 5.10, a izbornik na slici 5.11.

Uz to što je prikazan korisnikov tim, korisnik ima još opcija vezanih uz njega. Kao što se može vidjeti na slici 5.10, korisnik svakog pojedinačnog pokemona može ukloniti ili nasumično odabrati između 151 pokemona iz prve generacije, isto tako ima opciju obrisati cijeli tim ili ga cijelog nasumično odabrati.



Sl. 5.10. Početna stranica za registrinog korisnika

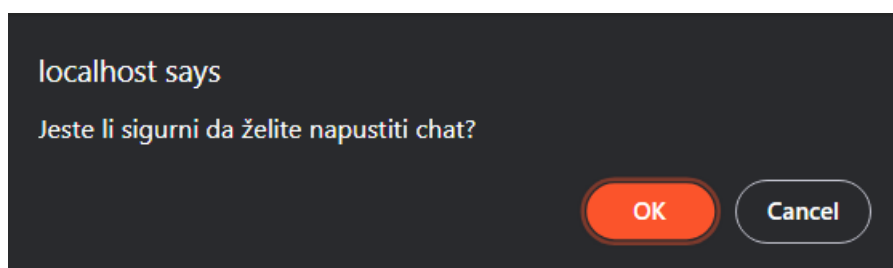


Sl. 5.11. Izbornik za registrinog korisnika

Prva dodatna stvar je „Chat“. U njemu prijavljeni korisnik može komunicirati s drugim prijavljenim korisnicima u stvarnom vremenu. Kada se korisnik pridruži chatu ili ga napusti, svi to vide. Sve navedeno je vidljivo na slici 5.12, ako korisnik odluči napustiti chat klikom na „Napusti Chat“ dobiti će poruku koju mora potvrditi kako bi napustio, ta poruka je vidljiva na slici 5.13.

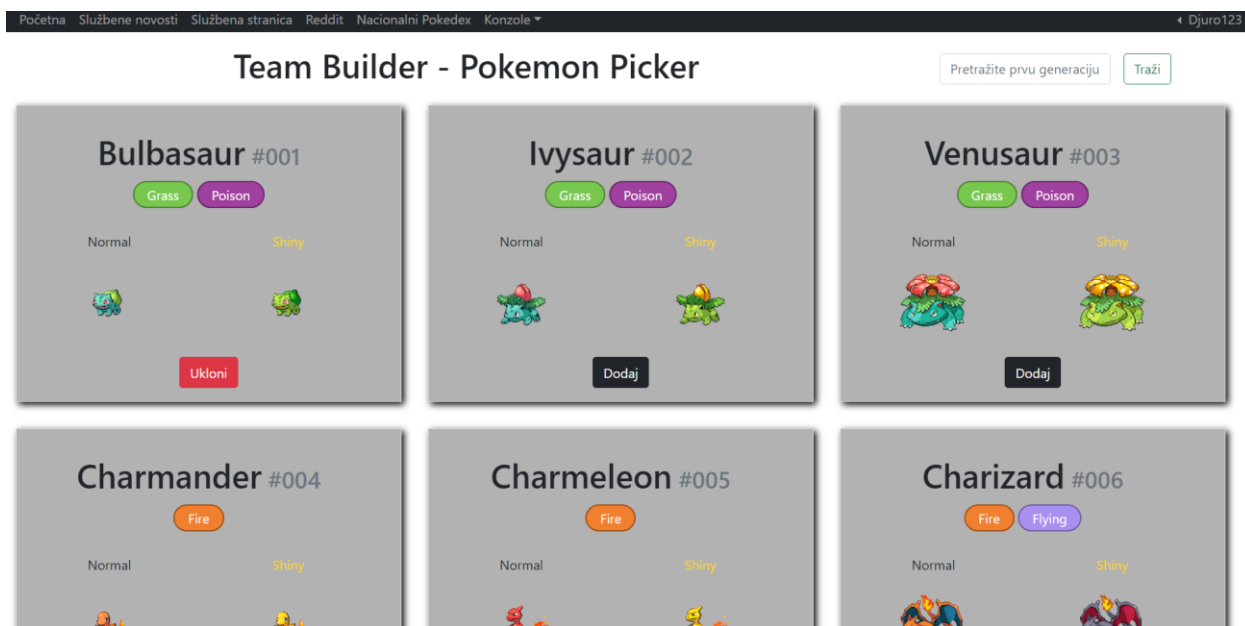


Sl. 5.12. Izgled Chat-a



Sl. 5.13. Potvrda za napuštanje Chat-a

Druga stvar koja je dostupna registriranim korisnicima je „Team Builder“ tj. izgradnja vlastitog tima od tri pokemona. Mogu birati između 151. pokemona iz prve generacije koji se nalaze u lokalnoj bazi. Na slici 5.14 može se vidjeti kako izgleda početna stranica kada se odaberi tim. Svaki pokemon imaju svoju „karticu“ na njoj su prikazane naziv, id, tip ili tipovi, slike za normalnu i shiny varijantu te gumb dodaj ili ukloni zavisi koliko i koji su pokemoni u timu. Ulaskom miša nad jednu od slika, slika se okrene. Ova izmjena slike prikazana je na slici 5.15. Klikom na naziv pokemona otvara se modal u kojemu se nalazi više informacija o pokemonu, to je vidljivo na slici 5.16. Još jedna stvar koja je dostupna na izgradnji tima je pretraživanje, pokemoni se mogu pretraživati po tipu, id-u i po njegovom nazivu. Rezultat pretrage se nalazi na slici 5.17.



Sl. 5.14. Izgled „Team Builder“ stranice



Sl. 5.15. Izgled slika na prijelaz miša

Mewtwo

Opis:

Its DNA is almost the same as Mew's. However, its size and disposition are vastly different.

Sposobnosti:

pressure, unnerve

Efektivan protiv:

Fighting Poison

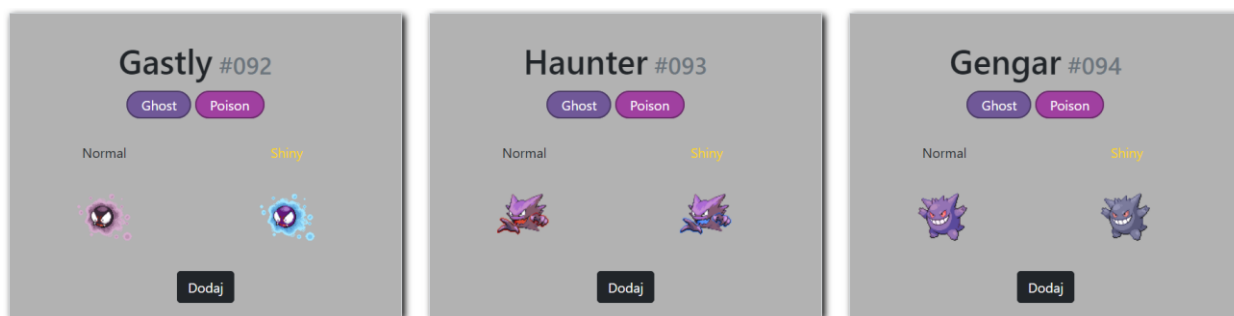
Slab protiv:

Bug Ghost Dark

HP:	ATT:	DEF:	SPA:	SPD:	SPEED:
106	110	90	154	90	130



Sl. 5.16. Detaljniji opis pokemona



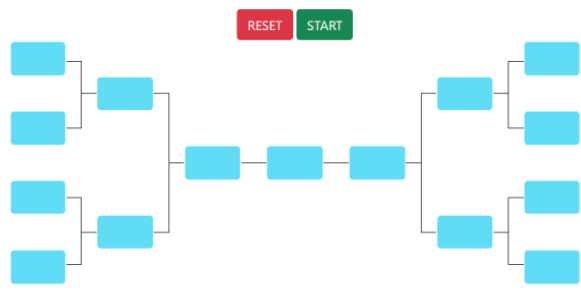
Sl. 5.17. Rezultat pretrage „Ghost“

Iduća stvar koju korisnik može koristiti je opcija za vođenje turnira. Turnir moraju igrati osam osoba kako bi se mogao pokrenuti, mogu sudjelovati svi prijavljeni korisnici. Stranica turnira se sastoji od tablice u kojoj su izlistani svi korisnici, njihov nadimak, tim, te gumb za odabiranje. Ispod tablice se nalazi izgled turnira u kojega se korisnici dodaju. Sve to je vidljivo na slici 5.18.

Početna Službene novosti Službena stranica Reddit Nacionalni Pokedex Konzole Djuro123

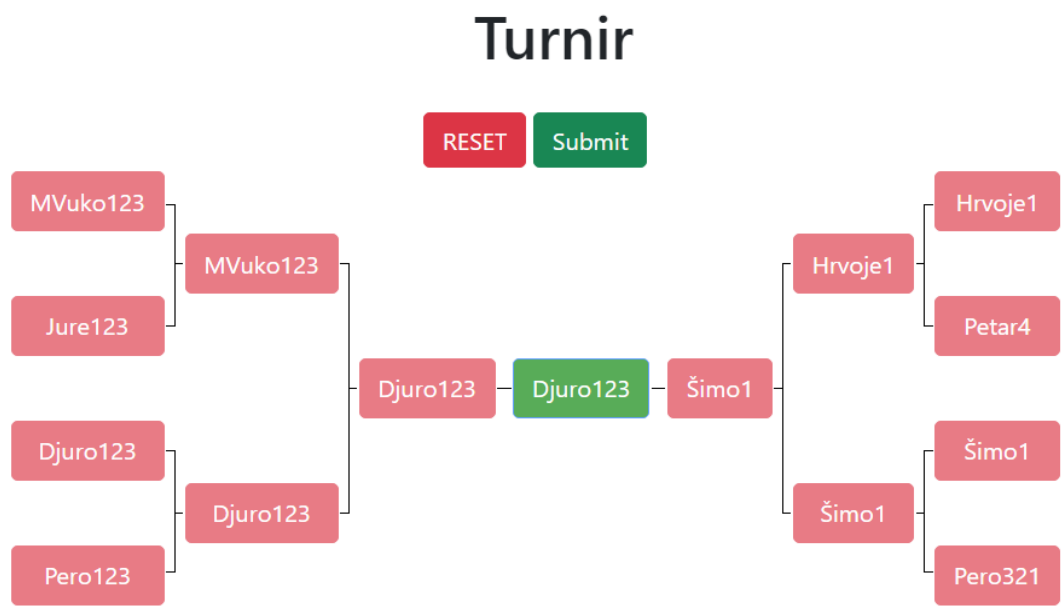
Turnir

Nadimak	Tim	Akcija
MVuko123	Venusaur, Gyarados, Mewtwo	<input type="button" value="Odaberi"/>
jure123	Bulbasaur, Charmander, Squirtle	<input type="button" value="Odaberi"/>
Djuro123	Bulbasaur, Charmeleon, Blastoise	<input type="button" value="Odaberi"/>



Sl. 5.18. Izgled „Turnir“ stranice

Kada je korisnik odabrao osam igrača za turnir, može kliknuti gumb start, a ako je nešto pogriješio u odabiru može kliknuti reset da sve napravi otpočetak. Kada je turnir krenuo sve se sakriva osim gumba reset i strukture turnira. Korisnik odabire tko je pobijedio u kojem dvoboju i tako skroz dok netko ne pobijedi. Kada je korisnik pobijedio na mjestu start gumba se pojavljuje submit gumb. Ove radnje su prikazane na slici 5.19.



Sl. 5.19. Izgled obavljenog turnira

Kada se klikne na gumb submit vidljiv na slici 5.19, vodi nas na novi ekran na kojemu je prikazana ljestvica svih korisnika koji imaju bodove, te broj bodova. Ljestvica je organizirana da na vrhu budu korisnici s najviše bodova, a na dnu s najmanje. Ljestvici se može pristupiti i iz izbornika koji je prikazan na slici 5.11. Izgled ljestvice se može vidjeti na slici 5.20.

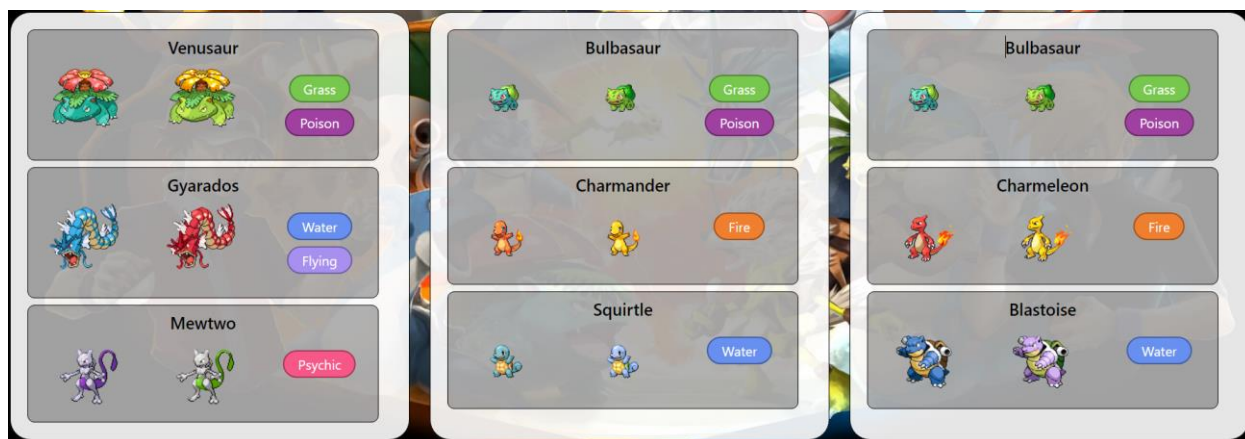
Nadimak	Bodovi
MVuko123	13.5
Šimo1	12
Petar4	9
Djuro123	7.5
Hrvoje1	6
Krešo1	3
Jure123	3
Pero123	1.5

Sl. 5.20. Izgled ljestvice

Zadnje što je ostalo je odjava. Klikom na odjavu u izborniku korisnik više neće biti registriran, imat će ograničeno što može raditi i bit će vraćen na početnu stranicu nebitno gdje je bio.

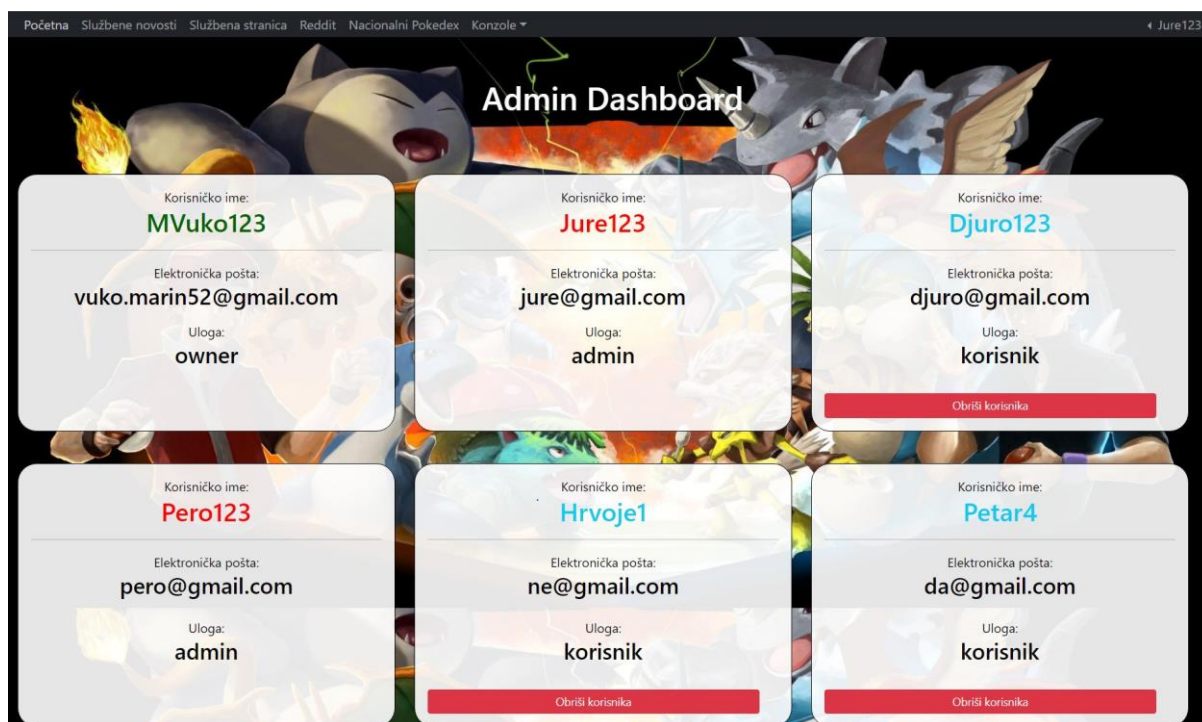
5.3. Admin i vlasnik

U aplikaciji postoji više uloga za registrirane korisnike. Sve gore navedeno u prethodnom poglavlju se odnosi na korisnike koji imaju ulogu korisnika, a za admina i vlasnika postoji još jedna stranica koja nije vidljiva ostalima, a to je „Dashboard“ tj. upravljačka ploča. Tu su vidljivi svi registrirani korisnici njihovo ime, e-mail i uloga, te na klik na njihovo ime pokazuje se druga strana „kartice“ na kojoj se nalazi njihov tim, to se može vidjeti na slici 5.21.

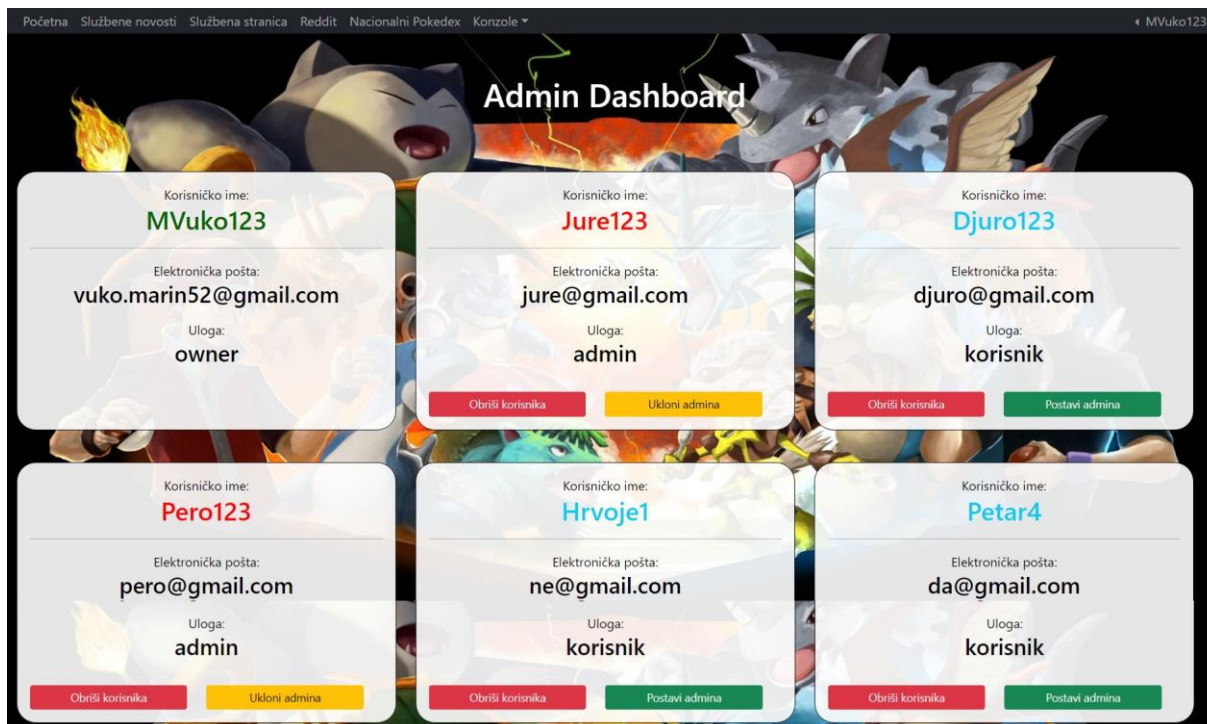


Sl. 5.21. Prikaz timova u „Dashboard“-u

Admin i vlasnik nemaju iste ovlasti. Admin može samo brisati korisnike, ali ne i druge admine, i njegov izgled stranice je kao i na slici 5.22, dok vlasnik ima jednu opciju više, a to je postavljanje ili brisanje admina što se može vidjeti na slici 5.23.



Sl. 5.22. Izgled „Dashboard“-a za admina



Sl. 5.23. Izgled „Dashboard“-a za vlasnika

6. Zaključak

U ovome završnome radu napravljen je web portal za kolekcionarsku kartašku igru. Aplikacija je izrađena s ciljem da korisnicima pruži sve dostupne informacije o svijetu pokemona na jednome mjestu, te da se mogu zabavljati i komunicirati s drugim ljudima o zajedničkom interesu. Postoje četiri uloge u aplikaciji: gost, korisnik, admin i vlasnik. Gost ima najmanje mogućnosti, jer nije registriran, te može samo pristupiti informacijama o igricama, konzolama i pokedexu. Korisnik je registriran, ima sve mogućnosti kao i gost, ali uz to ima i pristup chatu u kojemu može komunicirati sa svim prijavljenim korisnicima, izgradnji vlastitog tima od tri pokemona iz prve generacije, vidjeti ljestvicu s bodovima igrača, te može sudjelovati i voditi turnire. Admin i vlasnik uz sve što može korisnik još imaju pristup dashboardu u kojemu su vidljivi svi korisnici, njihove informacije i njihovi timovi. Obje uloge mogu obrisati korisnike, ali samo vlasnik može dodijeliti i maknuti admina. Korak prema unaprjeđenju stranice bi bio da su svi pokemoni dodani za stvaranje tima, a ne samo prva generacija. Turnir bi mogao biti unaprijeđen da postoji više opcija za igru, a ne samo opcija od osam igrača. Zadnja stvar koja bi se mogla ugraditi je opcija „borbe“ između prijavljenih korisnika, s njihovim timovima.

LITERATURA

- [1] https://bulbapedia.bulbagarden.net/wiki/Main_Page 12.06.2022
- [2] <https://www.serebii.net/> 12.06.2022
- [3] <https://marriland.com/> 12.06.2022
- [4] <https://www.thirdrocktechkno.com/blog/how-to-choose-a-technology-stack-for-web-application-development/> 21.06.2022
- [5] <http://www.webtech.com.hr/html.php> 21.06.2022
- [6] <https://krazytech.com/technical-papers/ajax-technology> 21.06.2022
- [7] <https://w3techs.com/technologies/details/cp-javascript> 05.07.2023
- [8] <https://haydenjames.io/78-of-the-web-powered-by-php-1-on-php-8/> 05.07.2023
- [9] „Stack Overflow Developer Survey 2021 Integrated Development Enviroment“
12.06.2022
- [10]“Usage statistics of JavaScript as client-side programming language on websites“
12.06.2022
- [11]“Usage statistics of PHP for websites“ 12.06.2022
- [12]<https://code.visualstudio.com/> 12.06.2022
- [13]https://en.wikipedia.org/wiki/Visual_Studio_Code 12.06.2022
- [14]<https://www.bu.edu/lernet/artemis/years/2020/projects/FinalPresentations/HTML/historyofhtml.html> 12.06.2022
- [15]https://www.washington.edu/accesscomputing/webd2/student/unit1/module3/html_history.html 12.06.2022
- [16]<https://en.wikipedia.org/wiki/HTML> 12.06.2022
- [17]<https://www.thirdrocktechkno.com/blog/how-to-choose-a-technology-stack-for-web-application-development/> 12.06.2022
- [18]<https://kenzie.snhu.edu/blog/front-end-vs-back-end-whats-the-difference/> 12.06.2022

- [19]<http://web.simmons.edu/~grabiner/comm244/weeknine/including-javascript.html>
12.06.2022
- [20]https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/What_is_JavaScript 12.06.2022
- [21]<https://en.wikipedia.org/wiki/JavaScript> 12.06.2022
- [22]<https://www.hackreactor.com/blog/what-is-javascript-used-for> 12.06.2022
- [23][https://en.wikipedia.org/wiki/Bootstrap_\(front-end_framework\)](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)) 12.06.2022
- [24]https://www.w3schools.com/bootstrap/bootstrap_get_started.asp 12.06.2022
- [25]<https://ageek.dev/css-evolution> 12.06.2022
- [26]<https://en.wikipedia.org/wiki/CSS> 12.06.2022
- [27]https://www.w3schools.com/css/css_intro.as 12.06.2022
- [28]https://developer.mozilla.org/en-US/docs/Learn/CSS/First_steps/What_is_CSS
12.06.2022
- [29]https://developer.mozilla.org/en-US/docs/Learn/CSS/Building_blocks/Values_and_units
12.06.2022
- [30]<https://developer.mozilla.org/en-US/docs/Glossary/property/CSS> 12.06.2022
- [31]https://developer.mozilla.org/en-US/docs/Glossary/CSS_Selector 12.06.2022
- [32]https://www.w3schools.com/html/html_css.asp 12.06.2022
- [33]https://www.w3schools.com/css/css_selectors.asp 12.06.2022
- [34]https://www.w3schools.com/css/css_combinators.asp 12.06.2022
- [35]https://www.w3schools.com/css/css_pseudo_classes.asp 12.06.2022
- [36]https://www.w3schools.com/css/css_pseudo_elements.asp 12.06.2022
- [37]https://www.w3schools.com/css/css_attribute_selectors.asp 12.06.2022
- [38][https://en.wikipedia.org/wiki/Ajax_\(programming\)](https://en.wikipedia.org/wiki/Ajax_(programming)) 12.06.2022
- [39]<https://t4tutorials.com/history-of-ajax/> 12.06.2022
- [40]https://www.w3schools.com/whatis/whatis_ajax.asp 12.06.2022

- [41]<https://t4tutorials.com/history-of-ajax/> 12.06.2022
- [42]https://www.w3schools.com/sql/sql_syntax.asp 12.06.2022
- [43]<https://en.wikipedia.org/wiki/SQL> 12.06.2022
- [44]<https://www.tutorialspoint.com/sql/index.htm> 12.06.2022
- [45]<https://infocenter.sybase.com/help/index.jsp?topic=/com.sybase.infocenter.dc38151.1540/doc/html/san1278452826146.htm> 12.06.2022
- [46]<https://www.educba.com/mysql-vs-mysqli/> 12.06.2022
- [47]https://en.wikipedia.org/wiki/PHP#cite_note-15 12.06.2022
- [48]<https://www.php.net/manual/en/intro-what-is.php> 12.06.2022
- [49]<http://www.netakademija.hr/sto-je-php/> 12.06.2022

SAŽETAK

Cilj ovog završnog rada je bio napraviti web aplikaciju za sve vezano uz Pokémon marku. Aplikacija korisnicima pruža pristup informaciji o svakoj generaciji pokemona, njihovim igricama i konzolama koje su korištene tada, pregled pokedexa sa svim pokemonima do kraja osme generacije, komunikacija s ostalim korisnicima, izgradnja svog tima i vođenje turnira. Na samom početku su opisane tehnologije koje su korištene za izradu web aplikacije. U radu je opisano kako pokrenuti aplikaciju na računalu. U aplikaciji postoje četiri uloge gost, korisnik, admin i vlasnik. Gost je ne registriran i ima ograničen pristup stranici, može imati samo pristup informacijama o igricama, konzolama i pokedexu. Korisnik je registriran i uz pristup informacijama kao i gost ima još i vođenje tima, chat, te turnir i ljestvicu. Admin i vlasnik imaju sve što ima i korisnik, te uz to imaju pristup dashboardu, u kojemu vide pregled svih korisnika i timova, mogu brisati korisnike, a vlasnik može postavljati i ukloniti admina.

Ključne riječi: baza podataka, kartaška igra, turnir, web aplikacija, web tehnologije

ABSTRACT

Title: Web portal for a collector's card game

The aim of this final project was to create a web application related to the Pokémon brand. The application provides users with access to information about each pokemon generation, their games, and the consoles used during that time. It also offers a pokedex overview of all the pokemon up until the end of the eighth generation, communication with other users, team building, and tournament management. The initial part of the project describes the technologies used to develop the web application. The document explains how to launch the application on a computer. The application consists of four user roles: guest, user, admin, and owner. A guest is unregistered and has limited access to the website, only being able to view information about the games, consoles, and Pokédex. A registered user has access to the same information as a guest, as well as the ability to manage their team, participate in chat, tournaments, and leaderboards. Admins and owners have all the privileges of a user, plus access to a dashboard where they can view an overview of all users and teams, delete users, and the owner can appoint or remove admins.

Keywords: card game, database, tournament, web application, web technologies