

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 000

Baza podataka i web-aplikacija za plivačka natjecanja

Ilan Vezmarović

Zagreb, lipanj 2023.

SADRŽAJ

1. Uvod	1
2. Specifikacija zahtjeva	2
2.1. Analiza zahtjeva	2
2.2. Obrasci uporabe	2
2.3. Opis obrazaca uporabe	4
3. Baza podataka	10
3.1. Korištene tehnologije	10
3.1.1. DBeaver	10
3.1.2. PostgreSQL	10
3.2. ER Model	11
3.2.1. Opis ER modela	11
3.2.2. Popis entiteta i atributa	12
3.2.3. Popis veza	13
3.3. Relacijski model	14
3.3.1. Preslikavanje ER modela u relacijski model baze podataka . .	14
3.3.2. Popis relacija	14
3.4. Stvaranje baze podataka	16
3.5. Popunjavanje baze podataka	18
4. Web aplikacija	21
4.1. Korištene tehnologije	21
4.1.1. Java Spring i JPA	21
4.1.2. Vue.js	21

4.1.3. Vuetify 3	22
4.2. MVC obrazac	22
4.3. Arhitektura web aplikacije	23
4.3.1. Sloj podataka	23
4.3.2. Sloj poslovne logike	24
4.3.3. Sloj korisničkog sučelja	26
5. Korisničke upute	27
5.1. Korisničko sučelje	27
5.1.1. Natjecanja	27
5.1.2. Kvalificirani plivači	28
5.1.3. Plivači	29
5.1.4. Bazeni	30
5.1.5. Svjetski rekordi	31
6. Zaključak	33
Literatura	34
Popis slika	35

1. Uvod

U današnjem dobu digitalne revolucije, postoji sve veća potreba za praćenjem i analizom sportskih rezultata. Plivanje je jedan od sportova za koji je izražena potreba za pohranom velikog broja informacija. Rezultat navedenog je velik broj sportskih aplikacija. Kod aplikacije vrlo bitno da bude jednostavna i intuitivna za korisničko korištenje.

Motivacija za izradu ove aplikacije proizlazi iz želje da se omogući plivačima, trenerima i drugim zainteresiranim korisnicima jednostavan i intuitivan pristup njihovim rezultatima. Kroz jednostavno korisničko sučelje, korisnici će moći brzo i lako pregledavati, analizirati i pratiti postignuća plivača na temelju plivačkih rezultata.

Za izgradnju ovakvog sustava bitne su dvije stvari. Prvo se treba definirati dobro strukturirana i skalabilna baza podataka u koju će se pohranjivati svi podaci. Druga bitna stvar je oblikovanje preglednog i jednostavnog korisničkog sučelja koje će korisnicima omogućiti ugodan pristup podacima.

U prvom dijelu rada navedena je specifikacija zahtjeva koje sustav treba ispuniti i njihova analiza. U drugom dijelu detaljno je opisano oblikovanje i struktura baze podataka. U trećem dijelu opisana je unutarnja implementacija web aplikacije, a u četvrtkom dijelu su priložene upute za korisnika.

2. Specifikacija zahtjeva

2.1. Analiza zahtjeva

Na početku izrade ovakvog sustava potrebno je definirati korisnike i njihove zahtjeve koje će sustav omogućavati. Primarni i jedini korisnik sustava je posjetitelj koji pregledava podatke o plivačkim aktivnostima.

Osnovni zahtjev posjetitelja je pregled i pretraživanje podataka o:

- plivačkim natjecanjima
- plivačkim rezultatima
- bazenima
- svjetskim rekordima
- limitima za nastupa na PH¹
- kvalificiranim plivačima za PH
- registriranim plivačima

Posjetitelj također ima opciju dodavanja novih natjecanja.

2.2. Obrasci uporabe

Modeliranje funkcionalnih zahtjeva sustava ostvareno je dijagramom obrazaca uporabe (engl. Use case diagram) izraženih kao UML dijagram s tekstualnim opisima obrazaca. Dijagram opisuje tipičnu uporabu sustava značajnu korisniku (aktoru) te prikazuje pogled na sustav koji naglašava njegovo vanjsko ponašanje prema korisniku (pogled interakcije). Na slici 2.1 prikazan je dijagram obrazaca uporabe.

¹Prvenstvo Hrvatske



Slika 2.1: Dijagram obazaca uporabe

2.3. Opis obrazaca uporabe

1. Pregledaj popis svih natjecanju

- **Glavni sudionik:** Korisnik.
- **Cilj:** Pregledati popis svih natjecanja.
- **Ostali sudionici:** Baza podataka.
- **Preduvjeti:** Uspostavljena veza s internetom, podaci upisani u bazu, poslužitelj dostupan
- **Željeni scenarij:** Korisnik odabire stranicu s prikazom natjecanja i uspješno pregledava popis svih natjecanja.
- **Mogući ostali scenariji:** Podaci o natjecanjima nisu uneseni u bazu, ne prikazuju se korisniku.

2. Pregledaj prikaz prijavljenih plivača po disciplini

- **Glavni sudionik:** Korisnik.
- **Cilj:** Pregledati popis broj prijavljenih plivača po disciplini.
- **Ostali sudionici:** Baza podataka.
- **Preduvjeti:** Uspostavljena veza s internetom, podaci upisani u bazu, poslužitelj dostupan.
- **Željeni scenarij:** Korisnik odabire ikonu s dijagramom na kartici s prikazom natjecanja i uspješno pregledava stupčasti dijagram s brojem prijavljenih plivača po disciplini.
- **Mogući ostali scenariji:** Podaci o natjecanjima nisu uneseni u bazu, ne prikazuju se korisniku.

3. Pregledaj detalje o natjecanju

- **Glavni sudionik:** Korisnik.
- **Cilj:** Pregledati detalje o pojedinom natjecanju.
- **Ostali sudionici:** Baza podataka.
- **Preduvjeti:** Uspostavljena veza s internetom, podaci upisani u bazu, poslužitelj dostupan.

- **Željeni scenarij:** Korisnik odabire karticu s prikazom natjecanja i uspješno pregledava detalje o pojedinom natjecanju.
- **Mogući ostali scenariji:** Podaci o natjecanjima nisu uneseni u bazu, ne prikazuju se korisniku.

4. Dodaj novo natjecanje

- **Glavni sudionik:** Korisnik.
- **Cilj:** Dodati novo natjecanje na popis svih natjecanja.
- **Ostali sudionici:** Baza podataka.
- **Preduvjeti:** Uspostavljena veza s internetom, podaci upisani u bazu, poslužitelj dostupan.
- **Željeni scenarij:** Pritiskom na gumb s plusom, prikazuje se obrazac za dodavanje novog natjecanja. Korisnik popunjava podatke o novom natjecanju. Korisnik odabire opciju *Save*. Novo natjecanje se prikazuje na popisu svih natjecanja
- **Mogući ostali scenariji:** Uneseni podaci su neispravni ili uneseni u neispravnom formatu. Korisniku se prikazuje poruka o grešci prilikom validacije

5. Pregledaj popis svih disciplina održanih na natjecanju

- **Glavni sudionik:** Korisnik.
- **Cilj:** Pregledati popis svih disciplina održanih na natjecanju.
- **Ostali sudionici:** Baza podataka.
- **Preduvjeti:** Uspostavljena veza s internetom, podaci upisani u bazu, poslužitelj dostupan.
- **Željeni scenarij:** Pritiskom na ikonu s dokumentom na kartici odabranom natjecanja korisnika se preusmjerava na stranicu o odabranom natjecanju. Korisniku se prikazuju podaci o odabranom natjecanju i popis održanih disciplina.
- **Mogući ostali scenariji:** Rezultati natjecanja nisu uneseni u bazu, ne prikazuju se korisniku.

6. Pregledaj listu rezultata po disciplini

- **Glavni sudionik:** Korisnik.
- **Cilj:** Pregledati poredak plivača po disciplini
- **Ostali sudionici:** Baza podataka.
- **Preduvjeti:** Uspostavljena veza s internetom, podaci upisani u bazu, poslužitelj dostupan.
- **Željeni scenarij:** Pritiskom na pojedinu disciplinu prikazuje se gumb *Event Results*. Klikom na prikazani gumb otvara se dijalog s poretком plivača u odabranoj disciplini na odabranom natjecanju.
- **Mogući ostali scenariji:** Rezultati natjecanja nisu uneseni u bazu, ne prikazuju se korisniku.

7. Pregledaj sve kvalificirane plivače

- **Glavni sudionik:** Korisnik.
- **Cilj:** Pregledati sve kvalificirane plivače za Državno PH 2023.
- **Ostali sudionici:** Baza podataka.
- **Preduvjeti:** Uspostavljena veza s internetom, podaci upisani u bazu, poslužitelj dostupan.
- **Željeni scenarij:** Korisnik odabire karticu s prikazom kvalificiranih plivača za Državno PH. Korisniku se prikazuje lista svih plivača koji su otplivali zadane limite.
- **Mogući ostali scenariji:** Limiti ili plivači nisu uneseni u bazu, ne prikazuju se korisniku.

8. Poredaj kvalificirane plivače prema stupcima

- **Glavni sudionik:** Korisnik.
- **Cilj:** Poredati kvalificirane plivače prema stupcima.
- **Ostali sudionici:** Baza podataka.
- **Preduvjeti:** Uspostavljena veza s internetom, podaci upisani u bazu, poslužitelj dostupan.

- **Željeni scenarij:** Korisnik pritiskom na pojedini stupac odabire različite mogućnosti poretka.
- **Mogući ostali scenariji:** Podaci nisu uneseni u bazu, ne prikazuju se korisniku.

9. Prikaži limite po disciplinama

- **Glavni sudionik:** Korisnik.
- **Cilj:** Pregledati limiti po disciplinama.
- **Ostali sudionici:** Baza podataka.
- **Preduvjeti:** Uspostavljena veza s internetom, podaci upisani u bazu, poslužitelj dostupan.
- **Željeni scenarij:** Korisnik se pritiskom na gumb *Qualifying Standards* na dnu stranicu prikazuje popis svih limita u dvije liste: jedna prikazuje limite u muškoj kategoriji, a druga limite u ženskoj kategoriji.
- **Mogući ostali scenariji:** Podaci nisu uneseni u bazu, ne prikazuju se korisniku.

10. Pretraži registrirane plivače

- **Glavni sudionik:** Korisnik.
- **Cilj:** Pretražiti registrirane plivače.
- **Ostali sudionici:** Baza podataka.
- **Preduvjeti:** Uspostavljena veza s internetom, podaci upisani u bazu, poslužitelj dostupan.
- **Željeni scenarij:** Korisnik odabire stranicu s prikazom polja za pretraživanje plivača. Korisniku unosi puno ime i prezime željenog plivača i odabire opciju prikaza pritiskom na gumb *Search*. Korisniku se prikazuje stranica s prikazom detalja o željenom plivaču.
- **Mogući ostali scenariji:**
 - (a) Uneseno ime i prezime ne postoji u bazi, korisnika se ne preusmjerava na stranicu o plivaču.

(b) Korisnik nema rezultata u bazi, ne prikazuje se lista najboljih rezultata.

11. Pretraži podatke o registriranom plivaču

- **Glavni sudionik:** Korisnik.
- **Cilj:** Pregledati informacije o registriranom plivaču
- **Ostali sudionici:** Baza podataka.
- **Preduvjeti:** Uspostavljena veza s internetom, podaci upisani u bazu, poslužitelj dostupan. Korisnik je unio ispravno ime i prezime plivača na stranici za pretraživanje registriranih plivača.
- **Željeni scenarij:** Korisniku se prikazuje stranica s informacijama o željenom plivaču i listom njegovih najboljih rezultata u pojedinoj disciplini.
- **Mogući ostali scenariji:** Podaci o plivaču nisu uneseni u bazu, ne prikazuju se njegovi najbolji rezultati.

12. Pregledaj popis bazena

- **Glavni sudionik:** Korisnik.
- **Cilj:** Pregledati popis svih bazena.
- **Ostali sudionici:** Baza podataka.
- **Preduvjeti:** Uspostavljena veza s internetom, podaci upisani u bazu, poslužitelj dostupan.
- **Željeni scenarij:** Korisnik odabire stranicu s prikazom svih bazena. Korisniku se prikazuje popis bazena unesenih u bazu.
- **Mogući ostali scenariji:** Podaci nisu uneseni u bazu, ne prikazuju se korisniku.

13. Pregledaj detalje o bazenu

- **Glavni sudionik:** Korisnik.
- **Cilj:** Pregledati detalje o pojedinom bazenu.
- **Ostali sudionici:** Baza podataka.
- **Preduvjeti:** Uspostavljena veza s internetom, podaci upisani u bazu, poslužitelj dostupan.

- **Željeni scenarij:** Korisnik odabire karticu s prikazom bazena i uspješno pregledava detalje o odabranom bazenu.
- **Mogući ostali scenariji:** Podaci o bazenu nisu uneseni u bazu, ne prikazuju se korisniku.

14. Pregledaj listu svjetskih rekorda

- **Glavni sudionik:** Korisnik.
- **Cilj:** Pregledati listu trenutnih svjetskih rekorda.
- **Ostali sudionici:** Baza podataka.
- **Preduvjeti:** Uspostavljena veza s internetom, podaci upisani u bazu, poslužitelj dostupan
- **Željeni scenarij:** Korisnik odabire stranicu s prikazom svjetskih rekorda. Korisniku se prikazuju trenutni svjetski rekordi u pojedinoj disciplini u muškoj i ženskoj kategoriji.
- **Mogući ostali scenariji:** Podaci o svjetskim rekordima nisu uneseni u bazu, ne prikazuju se korisniku.

3. Baza podataka

3.1. Korištene tehnologije

U ovom projektu korištene su dvije ključne tehnologije, DBeaver i PostgreSQL baza podataka, koje su omogućile učinkovito upravljanje podacima.

3.1.1. DBeaver

DBeaver je popularan alat za upravljanje bazama podataka koji pruža intuitivno korisničko sučelje i podršku za različite baze podataka, uključujući PostgreSQL. Odabir DBeavera omogućio je jednostavno povezivanje s bazom podataka, brzo izvršavanje upita i efikasno upravljanje shemom baze podataka.

3.1.2. PostgreSQL

PostgreSQL je otvorena i objektno-relacijska baza podataka koja je odabrana za ovaj projekt iz nekoliko razloga. Prvo, PostgreSQL pruža napredne mogućnosti za upravljanje podacima, uključujući podršku za kompleksne upite, transakcije i indeksiranje. Također nudi visoku razinu pouzdanosti, što je važno za ovu aplikaciju za praćenje plivačkih rezultata.

3.2. ER Model

ER model (engl. Entity-relationship model) baze podataka je konceptualni alat koji nam pomaže organizirati i prikazati strukturu podataka na intuitivan način. U ER modelu koristimo entitete koji predstavljaju stvari ili koncepte u stvarnom svijetu, kao što su korisnici, proizvodi ili narudžbe. Svaki entitet ima svoje atribute koji opisuju karakteristike tog entiteta, poput imena, adrese ili cijene. Osim toga, ER model koristi veze između entiteta kako bi prikazao njihove međusobne odnose, kao što su veze "ima" ili "pripada". Korištenjem ER modela, možemo bolje razumjeti strukturu podataka.

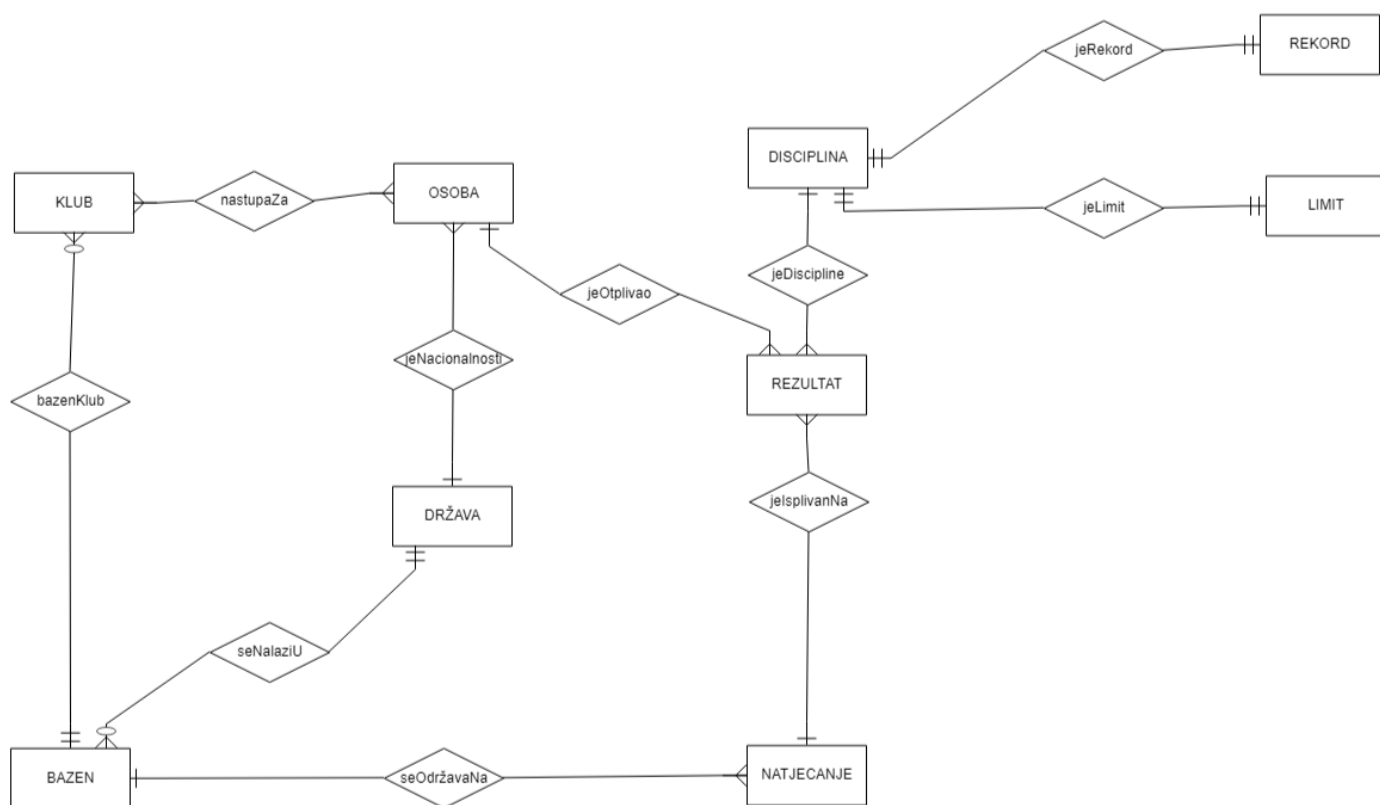
3.2.1. Opis ER modela

Podaci spremljeni u ovoj bazi uključuju informacije o plivačima(osobama), plivačkim natjecanjima, plivačkim bazenima, plivačkim rezultatima, limitima i svjetskim rekordima.

Središnji entitet ove baze je entitet rezultat jer se cijeli prikaz podataka temelji na rezultatima. Entitet rezultat predstavlja jedinstveno isplivano vrijeme određenog plivača u određenoj disciplini na određenom natjecanju. Zbog toga su veze *jeOtplivao*, *jeDiscipline* i *jeIsplivanNa* oblika 1:N.

Za plivača se uz osnovne podatke osobi bilježi i država koju predstavlja te klub za koji nastupa. Klub ima prijavljen matični bazen na kojemu održava svoje treninge. Bazen uz osnovne podatke o bazenu ima i podatke o mjestu na kojem se nalazi, tj. Državi u kojoj se nalazi. Natjecanje se održava na točno određenom bazenu.

Discipline su sve moguće pojedinačne discipline u muškoj i ženskoj kategoriji. Za svaku disciplinu se bilježi trenutni svjetski rekord i limit koji je potrebno isplivati za nastup na Državnom PH 2023. ER model je prikazan na slici 3.1.



Slika 3.1: ER model baze podataka

3.2.2. Popis entiteta i atributa

OSOBA - idosoba, imeosoba, prezimeosoba, spol, datumrodjenja, visina, tezina, idklub, iddržava

REZULTAT - idrezultat, vrijeme, bodovi, datum, idosoba, iddisciplina, idnatjecanje

NATJECANJE - idnatjecanje, nazivnatjecanje, vrstanatjecanje, datumod, datumdo, idbazen

BAZEN - idbazen, nazivbazen, kapacitet, grad, adresa, iddržava

KLUB - idklub, nazivklub, godinaosnivanja, idbazen

DISCIPLINA - iddisciplina, nazivdisciplina, spol

DRZAVA - iddržava, nazivdržava

LIMIT - idlimit, vrijeme, iddisciplina

REKORD - idrekord, vrijeme, iddisciplina

Napomena: Primarni ključevi podvučeni su punom linijom.

3.2.3. Popis veza

jeOtplivao - idrezultat, idosoba

jeIsplivanNa - idrezultat, idnatjecanje

nastupaZa - idosoba, idklub

jeNacionalnosti - idosoba, iddržava

bazenKlub - idbazen, idklub

seNalaziU - idbazen, iddržava

jeDiscipline - idrezultat, iddisciplina

jeLimit - iddisciplina, idlimit

jeRekord - iddisciplina, idrekord

Napomena: Primarni ključevi podvučeni su punom linijom.

3.3. Relacijski model

3.3.1. Preslikavanje ER modela u relacijski model baze podataka

Prikazani ER model preslikat će se u relacijski model PostgreSQL baze podataka. Strani ključevi nastaju na temelju N:1 prikazanih veza u kojima se primarni ključ sastoji od jednog atributa. Relacijski model baze podataka prikazan je na slici 3.2.

3.3.2. Popis relacija

Relacijski model sastoji se od navedenih relacija:

- **OSOBA** - idosoba, imeosoba, prezimeosoba, spol, datumrođenja, visina, težina, idklub, iddržava
Relacija 'Osoba' sadrži primarni ključ koji jednoznačno identificira svaku osobu. U relaciju se uz 'idosoba' spremaju: ime osobe, prezime osobe, spol, datum rođenja, visina i težina. Atribut 'idklub' je strani ključ koji referencira relaciju 'Klub' te predstavlja klub za koji plivač nastupa. Atribut 'iddržava' je strani ključ koji referencira relaciju 'Država' te predstavlja državu koju plivač predstavlja na natjecanjima.
- **REZULTAT** - idrezultat, vrijeme, bodovi, datum, idosoba, iddisciplina, idnatjecanje, idnatjecanje, idosoba
Relacija 'Rezultat' sadrži primarni ključ koji jednoznačno identificira svaki rezultat pohranjen u bazi. U relaciju se spremaju vrijeme rezultata, bodovi (izračunati u odnosu na svjetski rekord), datum na koji je rezultat isplivan. Relacija sadrži tri strana ključa. Atribut 'idosoba' je strani ključ koji referencira relaciju 'Osoba' i predstavlja osobu koja je isplivala rezultat. Atribut 'iddisciplina' je strani ključ koji referencira relaciju 'Disciplina' i predstavlja disciplinu u kojoj je rezultat isplivan. Atribut 'idnatjecanje' je strani ključ koji referencira relaciju 'Natjecanje' i predstavlja natjecanje na kojem je rezultat isplivan.

- **NATJECANJE** - idnatjecanje, nazivnatjecanje, vrstanatjecanje, datumod, datumdo, idbazen

Relacija 'Natjecanje' sadrži primarni ključ 'idnatjecanje' koji jednoznačno identificira svako natjecanje. Ostali atributi u relaciji su naziv natjecanja, vrsta natjecanja - koja se odnosi na olimpijske igre, svjetsko prvenstvo i sl. Uz ove attribute bilježi se datum početka i datum završetka natjecanja. Atribut 'idbazen' je strani ključ koji referencira relaciju 'Bazen' i predstavlja bazen na kojem se natjecanje održava.

- **BAZEN** - idbazen, nazivbazen, kapacitet, grad, adresa, iddržava

Relacija 'Bazen' predstavlja registrirane olimpijske bazene i sadrži primarni ključ 'idbazen'. U relaciju se pohranjuje naziv bazena, kapacitet ljudi koji mogu pristupiti bazene te grad i adresa na kojoj se bazen nalazi. Atribut 'iddržava' je strani ključ koji predstavlja državu u kojoj se bazen nalazi i referencira relaciju 'Država'.

- **KLUB** - idklub, nazivklub, godinaosnivanja, idbazen

Relacija 'Klub' predstavlja prijavljene klubove koji sudjeluju na natjecanjima. Atribut 'idklub' je primarni ključ. Ostali atributi u relaciji predstavljaju podatke od nazivu klub i godini osnivanja kluba. Atribut 'idbazen' je strani ključ koji referencira relaciju 'Bazen' i predstavlja matični bazen kluba na kojemu se održavaju treninzi.

- **DISCIPLINA** - iddisciplina, nazivdisciplina, spol

U relaciju 'Disciplina' pohranjuju se podaci o svim mogućim pojedinačnim disciplinama, dakle 50m, 100m, 200m u svakom stilu (slobodno, leđno, prsno i leptir) i posebno u muškoj i ženskoj kategoriji te 400, 800 i 1500 slobodno, 200 i 400 mješovito. Dakle atribut 'iddisciplina' je primarni ključ, a naziv discipline je jedan od navedenih u kombinaciji sa spolom.

- **DRZAVA** - iddrzava, nazivdrzava

U relaciji 'Država' pohranjuju se nazivi države uz primarni ključ na osnovi atributa 'iddrzava' koji je primarni ključ.

- **LIMITI** - idlimit, vrijeme, iddisciplina

U relaciji 'Limiti' pohranjuju se limiti koje je potrebno isplivati za nastup na Državnom PH 2023. Atribut 'idlimit' je primarni ključ, a 'iddisciplina' je strani ključ koji referencira relaciju 'Disciplina'. Vrijeme je vrijeme limita.

- **REKORD** - idrekord, vrijeme, iddisciplina

U relaciji 'Rekord' pohranjuju se svjetski rekordi, dakle najbrža vremena u svakoj disciplini u muškoj i ženskoj kategoriji. Atribut 'idlimit' je primarni ključ, a 'iddisciplina' je strani ključ koji referencira relaciju 'Disciplina'. Vrijeme je vrijeme svjetskog rekorda.

Napomena: Primarni ključevi podvučeni su punom linijom, a strani ključevi isprekidanom linijom.

3.4. Stvaranje baze podataka

U nastavku slijede neke od SQL naredbi za kreiranje relacija baze podataka:

```
CREATE TABLE IF NOT EXISTS osoba (
    idosoba SERIAL PRIMARY KEY,
    imeosoba VARCHAR,
    prezimeosoba VARCHAR,
    spol VARCHAR,
    datumrodjenja DATE,
    visina NUMERIC(5, 2),
    tezina NUMERIC(5, 2),
    idklub INT,
    iddrzava INT,
```

```
FOREIGN KEY (idklub) REFERENCES klub (idklub),  
FOREIGN KEY (iddrzava) REFERENCES drzava (iddrzava)  
);
```

```
CREATE TABLE rezultat (  
    idrezultat SERIAL PRIMARY KEY,  
    vrijeme TIMESTAMP(3) WITHOUT TIME ZONE,  
    bodovi INT,  
    datum DATE,  
    idosoba INT,  
    iddisciplina INT,  
    idnatjecanje INT,  
    FOREIGN KEY (idosoba) REFERENCES osoba (idosoba),  
    FOREIGN KEY (iddisciplina) REFERENCES disciplina (  
        iddisciplina),  
    FOREIGN KEY (idnatjecanje) REFERENCES natjecanje (  
        idnatjecanje)  
);
```

```
CREATE TABLE natjecanje (  
    idnatjecanje SERIAL PRIMARY KEY,  
    nazivnatjecanje VARCHAR,  
    vrstanatjecanje VARCHAR,  
    datumod DATE,  
    datumdo DATE,  
    FOREIGN KEY (idbazen) REFERENCES bazen (idbazen)  
    idbazen INT,  
);
```

3.5. Popunjavanje baze podataka

U nastavku slijede neke od INSERT SQL naredbi za popunjavanje baze podataka podacima.

```
INSERT INTO osoba (idosoba, iddrzava, imeosoba,  
    prezimeosoba, spol, datumrodjenja, visina, tezina)  
VALUES
```

```
( (SELECT iddrzava FROM drzava WHERE nazivdrzava = '  
    Croatia'), 'Leo', 'Sudar', 'M', '1998-02-21', 182,  
    81, 4),  
( (SELECT iddrzava FROM drzava WHERE nazivdrzava = '  
    Croatia'), 'Ivano', 'Petrovic', 'M', '1995-11-14',  
    185, 98, 9),  
( (SELECT iddrzava FROM drzava WHERE nazivdrzava = '  
    Croatia'), 'Lovro', 'Tomic', 'M', '1998-09-09',  
    168, 89, 1),  
( (SELECT iddrzava FROM drzava WHERE nazivdrzava = '  
    Croatia'), 'Nino', 'Lovric', 'M', '1992-06-27',  
    170, 91, 9),  
( (SELECT iddrzava FROM drzava WHERE nazivdrzava = '  
    Croatia'), 'Marko', 'Radic', 'M', '2000-09-24',  
    177, 82, 6),  
( (SELECT iddrzava FROM drzava WHERE nazivdrzava = '  
    Croatia'), 'Dario', 'Peric', 'M', '2004-11-29',  
    181, 91, 9),
```

```
INSERT INTO rezultat (idrezultat, vrijeme, bodovi,  
    datum, iddisciplina, idnatjecanje, idosoba)  
VALUES  
(2055, 1, 15, '1970-01-01 00:00:29.419', 710, '
```

```

        2022-11-12', 97),
(2056, 1, 15, '1970-01-01 00:00:25.479', 820, '
        2022-11-12', 51),
(2057, 1, 15, '1970-01-01 00:00:25.622', 816, '
        2022-11-12', 133),
(2058, 1, 15, '1970-01-01 00:00:25.150', 831, '
        2022-11-12', 154),
(2059, 1, 15, '1970-01-01 00:00:27.246', 767, '
        2022-11-12', 106),
(2060, 1, 15, '1970-01-01 00:00:26.855', 778, '
        2022-11-12', 60),
(2064, 1, 15, '1970-01-01 00:00:29.847', 700, '
        2022-11-12', 143);

```

```

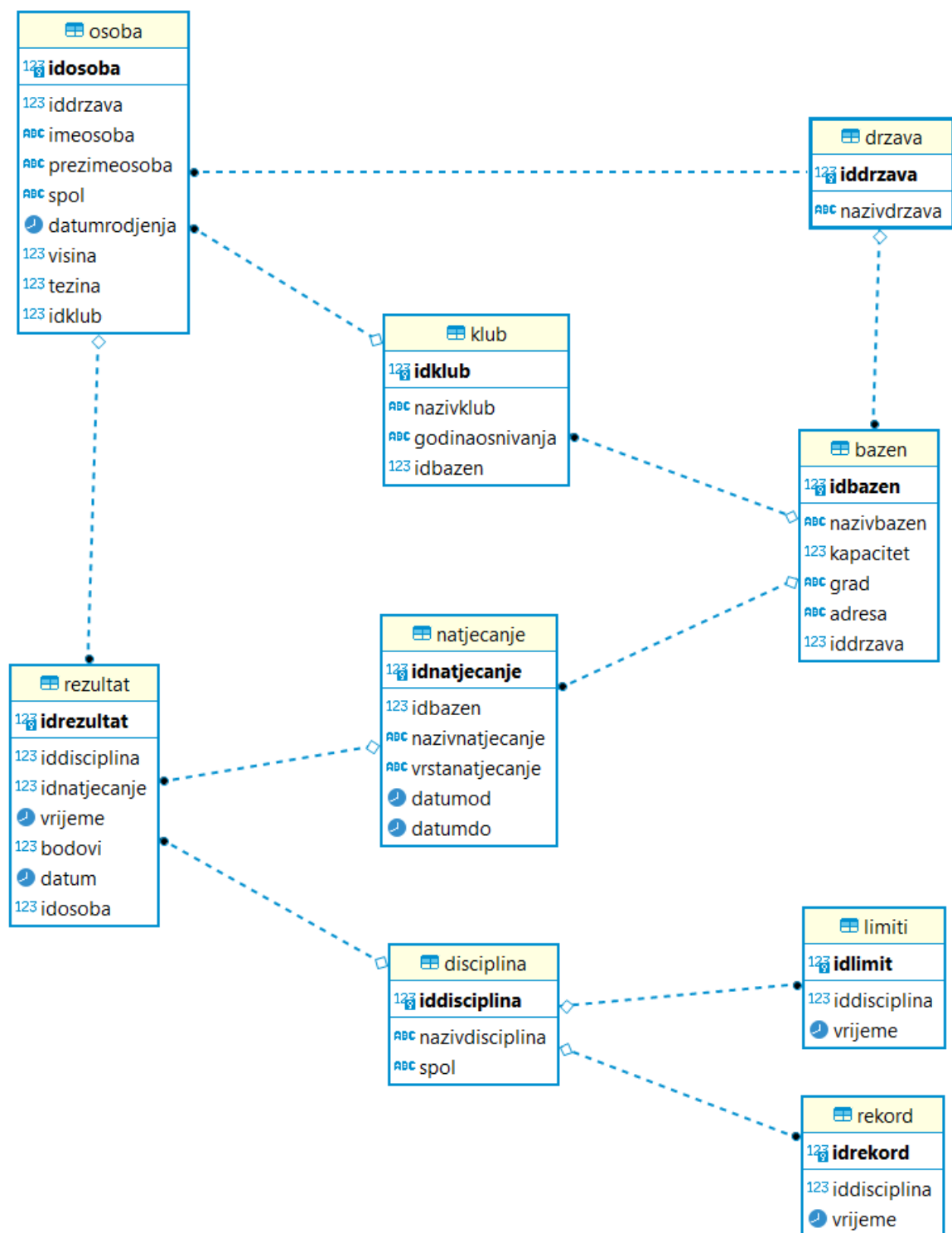
INSERT INTO rekordi (idrekord, iddisciplina, vrijeme)
VALUES

```

```

(1, 1, '00:00:20.910'),
(2, 2, '00:00:23.670'),
(3, 3, '00:00:46.860'),
(4, 4, '00:00:51.710'),
(5, 5, '00:01:42.000'),
(6, 6, '00:01:52.980'),
(7, 7, '00:03:40.070'),

```



Slika 3.2: Relacijski model baze podataka

4. Web aplikacija

4.1. Korištene tehnologije

4.1.1. Java Spring i JPA

Java Spring je popularni okvir za razvoj web aplikacija koji olakšava izgradnju pouzdanih i skalabilnih aplikacija. Koristeći koncepte poput inverzije upravljanja i ubrizgavanja ovisnosti, programeri mogu efikasno razvijati i testirati aplikacije. JPA (Java Persistence API) je standardna Java specifikacija koja omogućava upravljanje podacima u bazama podataka. Kombinacija Springa i JPA-e olakšava mapiranje Java objekata na bazu podataka i izvršavanje osnovnih operacija nad podacima.

4.1.2. Vue.js

Vue.js je moderni JavaScript okvir koji se koristi za izgradnju korisničkih sučelja te pruža modularnost, reaktivnost i jednostavnu integraciju s drugim alatima. Uspoređujući s običnim JavaScriptom, Vue.js omogućava brži razvoj aplikacija s manje koda i pruža bogat ekosustav resursa za podršku.

Sastoji se od tri glavna dijela: HTML, CSS i JavaScript.

- **HTML:** Vue.js koristi HTML za definiranje strukture korisničkog sučelja. Korisnik definira elemente kao što su gumbi i forme koji će se prikazivati.
- **CSS:** CSS se koristi za definiranje izgleda i stila elemenata u Vue.js aplikacijama. Korisnik može primijeniti različite stilove, boje, margine i druge stilizacijske atribute kako bi prilagodio izgled korisničkog sučelja.
- **JavaScript:** JavaScript je jezik koji se koristi za definiranje interaktivnosti i lo-

gike u Vue.js aplikacijama. Korisnik može koristiti JavaScript kako bi reagirao na događaje korisnika, upravljao podacima i izvršavao različite operacije.

4.1.3. Vuetify 3

Vuetify je napredna biblioteka komponenti za Vue.js koja olakšava izradu modernih korisničkih sučelja. Sadrži bogat skup gotovih komponenti kao što su tipografija, gumbi, forme, tablice i mnoge druge. Koristi se za izradu intuitivnog dizajna i popularan je za izgradnju atraktivnih i responzivnih web aplikacija.

4.2. MVC obrazac

MVC (Model-View-Controller) arhitektura je popularan koncept koji se koristi u razvoju aplikacija kako bi se postigla bolja organizacija i razdvajanje odgovornosti. Ova arhitektura se temelji na ideji podjele aplikacije na tri glavna dijela: Model, Pogled i Nadglednik. Svaki od tih dijelova ima specifičnu ulogu i zajedno stvaraju cjelovitu aplikaciju.

MVC arhitektura omogućava bolje upravljanje i organizaciju aplikacija kroz sljedeće glavne značajke:

1. Model:

- Model predstavlja poslovne podatke i logiku aplikacije.
- Ovdje se čuvaju i obrađuju podaci te se izvršavaju operacije na njima.
- Model je odgovoran za interakciju s bazu podataka ili drugim izvorima podataka.

2. Pogled:

- Pogled predstavlja korisničko sučelje kroz koje korisnik komunicira s aplikacijom.
- Ovdje se prikazuju podaci iz Modela na korisnički prihvatljiv način.
- Pogled je odgovoran za prikazivanje podataka i reagiranje na korisničke akcije.

3. Nadglednik:

- Nadglednik upravlja logikom aplikacije i korisničkim zahtjevima.
- Prima ulazne podatke od korisnika putem Pogleda i obrađuje ih.
- Nadglednik ažurira Model s novim podacima i ažurira Pogled kako bi odražavao promjene.

Prednosti korištenja MVC arhitekture su:

- Jasnija organizacija i struktura koda, što olakšava čitanje i održavanje aplikacije.
- Bolja razdvojenost odgovornosti između dijelova aplikacije, što olakšava izmjene i proširenja.
- Lakše testiranje jer se logika i prikaz odvajaju.
- Omogućava paralelni razvoj, jer timovi mogu istovremeno raditi na različitim dijelovima aplikacije.
- Poboljšava ponovnu upotrebu koda, jer se logika može koristiti s različitim prikazima.

Ukratko, MVC arhitektura pruža strukturu i organizaciju za razvoj aplikacija, olakšava održavanje, testiranje i proširenje.

4.3. Arhitektura web aplikacije

Web aplikacija za prikaz plivačkih rezultata i pregledavanje podataka o plivačkim natjecanjima koristi svoj MVC obrazac za organizaciju arhitekture.

4.3.1. Sloj podataka

Sloj podataka sastoji se od dva dijela: razred Model (ili Entitet) i sučelje Repository.

1. Model

Za svaku relaciju u bazi podataka mora postojati određeni entitet u sloju podataka koji predstavlja tu relaciju i njene atribute. Pomoću tog modela vrši se

mapiranje između privatnih članskih varijabli razreda Modela i atributa u tablici. Odsječak koda na slici 4.1 prikazuje model Natjecanje. Strani ključ je modeliran referencom na cijeli razred referencirajuće relacije.

```
@Entity
@Table(name = "natjecanje")
@Data
public class Natjecanje implements Serializable{

    /**
     *
     */
    private static final long serialVersionUID = 1L;

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name = "idnatjecanje")
    private Long idnatjecanje;

    @ManyToOne
    @JoinColumn(name = "idbazen", referencedColumnName = "idbazen")
    private Bazen bazen;

    @Column(name = "nazivnatjecanje")
    private String nazivnatjecanje;

    @Column(name = "vrstanatjecanje")
    private String vrstanatjecanje;

    @Column(name = "datumod")
    private Date datumod;

    @Column(name = "datumdo")
    private Date datumdo;
}
```

Slika 4.1: Razred Natjecanje - model

2. Repozitorij

Za svaki entitet definira se sučelje Repository koje nasljeđuje sučelje JpaRepository te time nasljeđuje i metode za dohvat podataka iz baze. JPA omogućuje pisanje SQL upita za dohvat podataka koristeći jednostavnu metodu imenovanja kojom u imenu metode definiramo po kojim atributima želimo da se upit vrši. Npr. Ako želimo dohvatiti natjecanje prema jedinstvenom 'idnatjecanje' jednostavno definiramo metodu `List<Natjecanje> findByIdnatjecanje(Long idnatjecanje)` (prikazano na slici 4.2) koju koristimo u Kontroleru.

4.3.2. Sloj poslovne logike

Kontroler (engl. Controller) je sloj koji se definira za svaki entitet Model i on obrađuje ulazne HTTP zahtjeve. Ovisno o dolaznom zahtjevu odabire metodu koju će pozvati

```

@Repository
public interface NatjecanjeRepository extends JpaRepository<Natjecanje, Long> {

    List<Natjecanje> findByIdnatjecanje(Long idnatjecanje);

    List<Natjecanje> findAllByOrderByDatumodDesc();

    @Query("SELECT DISTINCT n.vrstanatjecanje FROM Natjecanje n WHERE n.vrstanatjecanje <> ''")
    List<String> findDistinctCompetitionTypes();
}

```

Slika 4.2: Sučelje NatjecanjeRepository - repozitorij

i tako vraća podatke. Na slici 4.3 vidimo primjer Java Kontrolera s prikazom metoda koje se nalaze svaka na svojoj putanji.

```

@RestController
@RequestMapping("/natjecanja")
public class NatjecanjeController {

    private final NatjecanjeRepository natjecanjeRepository;

    public NatjecanjeController(NatjecanjeRepository natjecanjeRepository) {
        this.natjecanjeRepository = natjecanjeRepository;
    }

    @GetMapping
    public ResponseEntity<List<Natjecanje>> getAllNatjecanje() {
        List<Natjecanje> natjecanjeList = natjecanjeRepository.findAllByOrderByDatumodDesc();
        return ResponseEntity.ok(natjecanjeList);
    }

    @GetMapping("/{id}")
    public ResponseEntity<List<Natjecanje>> getNatjecanjeDetailsById(@PathVariable Long id) {
        List<Natjecanje> results = natjecanjeRepository.findByIdnatjecanje(id);
        return ResponseEntity.ok(results);
    }

    @PostMapping
    public ResponseEntity<Natjecanje> createNatjecanje(@RequestBody Natjecanje natjecanje) {
        Natjecanje createdNatjecanje = natjecanjeRepository.save(natjecanje);
        return ResponseEntity.status(HttpStatus.CREATED).body(createdNatjecanje);
    }

    @GetMapping("/vrste")
    public ResponseEntity<List<String>> getDistinctCompetitionTypes() {
        List<String> natjecanjeList = natjecanjeRepository.findDistinctCompetitionTypes();
        return ResponseEntity.ok(natjecanjeList);
    }
}

```

Slika 4.3: Razred NatjecanjeController - kontroler

Drugi dio poslovne logike je ostvaren u Vue.js JavaScriptu pomoću kojega se komunicira s kontrolerima na poslužiteljskoj strani. Na slici 4.4 je prikazan isječak koda u kojem se dohvaćaju podaci iz baze podataka na klijentsku stranu.

```
const actions = {
  fetchCompetitions({ commit }) {
    console.log('/api/natjecanja')
    axios
      .get('/natjecanja')
      .then(response => {
        console.log("responseData: " + response.data)
        commit('setCompetitions', response.data);
      })
      .catch(error => {
        console.error(error)
      })
  },
}
```

Slika 4.4: Vue.js akcija za dohvat podataka sa servera

4.3.3. Sloj korisničkog sučelja

Ovo je sloj koji komunicira s korisnikom. Korisnik ne mora znati ništa o sloju podataka, niti o sloju poslovne logike da bi uspješno koristio aplikaciju. Korisničko sučelje je izvedeno pomoću JavaScript radnog okvira Vue.js te se za gotove komponente i stilizacijske elemente koristi Vuetify biblioteka. Na slici 4.5 je prikazan isječak koda jedne komponente za prikaz podataka korisnicima.

```
<template>
  <Layout title="Swimming competitions">
    <v-container>
      <v-row>
        <v-col v-for="(competition, index) in competitions" :key="index" cols="12" sm="6" md="4" lg="3">
          <div v-if="competition.nazivnatjecanje!=null">
            <CompetitionCard
              :competition="competition"
            >
          </div>
        </v-col>
        <v-col class="plus-container" cols="12" sm="6" md="4" lg="3">
          <AddCompetition />
        </v-col>
      </v-row>
    </v-container>
  </Layout>
</template>
```

Slika 4.5: Vue.js komponenta za prikaza kartica natjecanja

5. Korisničke upute

5.1. Korisničko sučelje

Korisničko sučelje omogućava pregled svih podataka o rezultatima, plivačima, limitima, bazenima i natjecanjima.

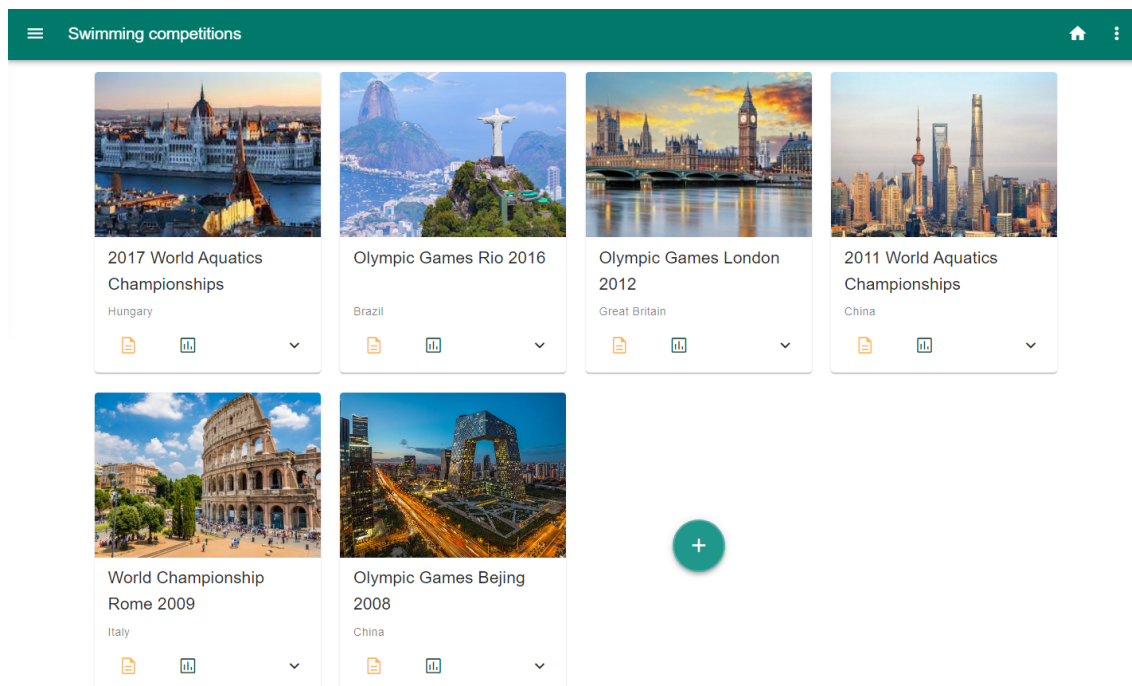
Nakon pokretanja aplikacije korisnik se nalazi na početnom zaslonu. Korisnik može odabrati druge stranice klikom na strelice na lijevom i desnom uglu stranice. Korisnik također može koristiti navigacijsku traku za navigiranje kroz aplikaciju koja se prikazuje klikom na gumb u gornjem desnom uglu stranice. Moguće opcije su redom: natjecanja, kvalificirani plivači, plivači, bazeni te svjetski rekordi.

5.1.1. Natjecanja

Odabirom kartice natjecanja prikazuju se sva registrirana natjecanja (slika 5.1). Korisnik na kartici za natjecanje može odabrati tri opcije:

1. Ikona dokumenta - prikaz rezultata s natjecanja. Na ovoj stranici potrebno je odabrati željenu disciplinu i pritisnuti gumb 'Rezultati discipline' da bi se prikazali svi rezultati u toj disciplini (slika 5.2).
2. Ikona grafa - prikaz stupčastog dijagrama prijavljenih plivača za natjecanje.
3. Strelica - prikaz detalja o natjecanju.

Na kraju liste prikaza natjecanja nalazi se gumb za dodavanje novog natjecanja. Korisniku se klikom na gumb prikazuje forma (slika 5.3) za dodavanje novog natjecanja.



Slika 5.1: Kartica s prikazom svih natjecanja



Slika 5.2: Kartica s prikazom detalja o odabranom natjecanju

5.1.2. Kvalificirani plivači

Odabirom kartice kvalificirani plivači prikazuje se popis kvalificiranih plivača i discipline u kojima su se kvalificirali. Na dnu stranice pritiskom na gumb (slika 5.4). Limiti prikazat će se limiti po disciplinama.

Slika 5.3: Forma za dodavanje novog natjecanja

Athletes qualified for National Championship 2023				
58	David	Novak	Igra	400m individual medley, 50m butterfly, 1500m freestyle, 100m breaststroke, 100m breaststroke, 400m individual medley
85	Ivan	Kušec	Mladost	400m freestyle, 50m butterfly, 100m butterfly
86	Filip	Perić	Dubrava	200m freestyle, 50m backstroke, 50m butterfly, 200m butterfly
151	Lovro	Pavić	Plivački klub Sisak Janaf	100m breaststroke, 50m butterfly, 100m breaststroke, 400m freestyle, 800m freestyle
177	Dinko	Knežić	Plivački klub Sisak Janaf	50m butterfly, 200m freestyle, 100m butterfly, 800m freestyle, 400m freestyle
124	Leo	Radoš	Kantrida	400m individual medley, 200m backstroke, 200m backstroke, 200m butterfly, 100m breaststroke, 800m freestyle, 50m butterfly
202	Roko	Lovrić	Jadran	50m butterfly, 200m backstroke, 100m breaststroke, 200m individual medley
48	Petar	Horvat	Jadran	200m individual medley, 200m breaststroke, 400m freestyle, 100m backstroke, 200m freestyle, 50m breaststroke, 50m butterfly, 1500m freestyle
56	Marin	Luketić	Plivački klub Sisak Janaf	50m butterfly, 800m freestyle, 200m backstroke, 100m backstroke, 800m freestyle
159	Stjepan	Luketić	Jug	50m butterfly, 100m butterfly
91	Stipe	Mlinarić	Jug	100m breaststroke, 50m butterfly, 200m breaststroke, 800m freestyle, 400m individual medley, 200m breaststroke
62	Marin	Marković	Mladost	800m freestyle, 100m breaststroke, 800m freestyle, 200m freestyle, 50m butterfly
162	Antonio	Luketić	Jug	200m freestyle, 400m freestyle, 100m backstroke, 100m butterfly, 200m butterfly, 400m freestyle, 50m butterfly, 800m freestyle

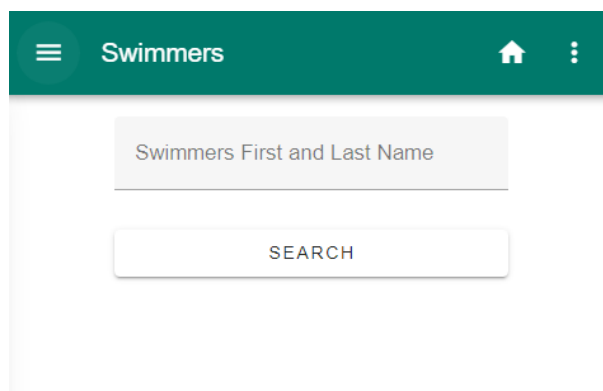
QUALIFYING TIME STANDARDS

Slika 5.4: Kartica s prikazom kvalificiranih plivača i limita

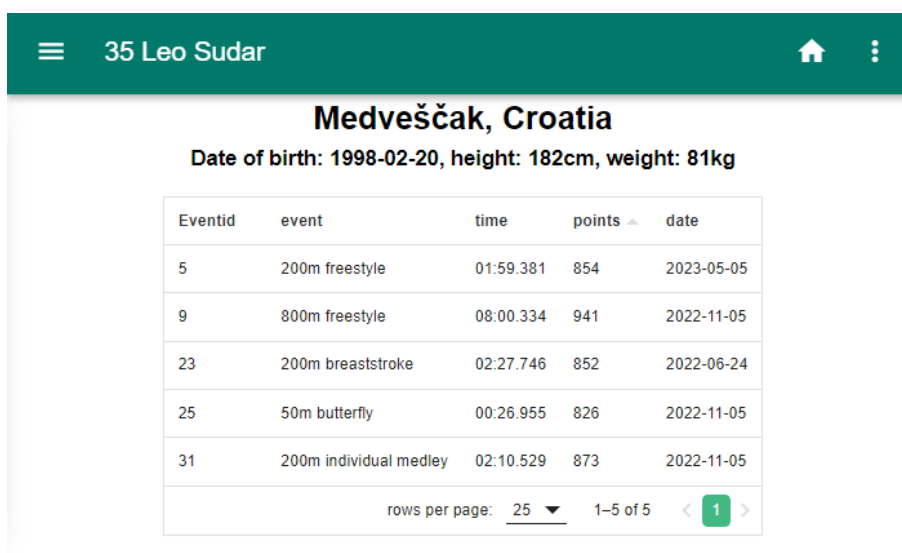
5.1.3. Plivači

Odabirom kartice plivači prikazuje se polje za unos teksta u koje korisnik treba unijeti ime traženog plivača (slika 5.5). Pritiskom na gumb Traži otvara se stranica s detaljima

o traženom plivaču i njegovim najboljim rezultatima (slika 5.6).



Slika 5.5: Kartica za pretraživanje plivača

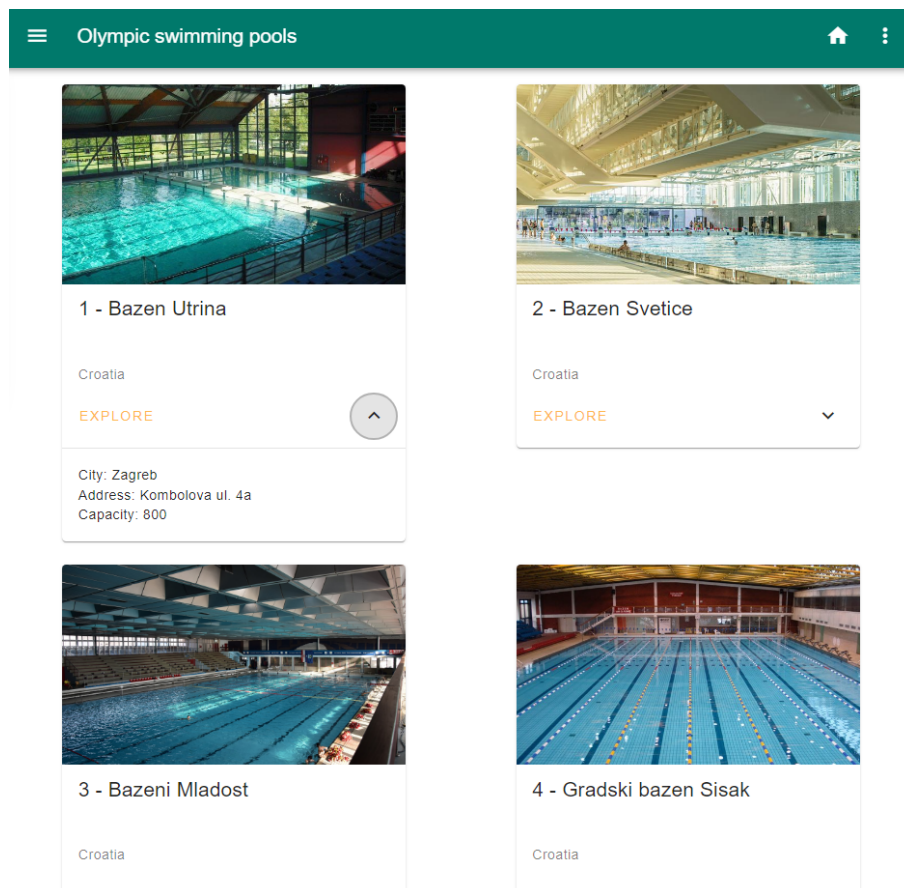


Eventid	event	time	points	date
5	200m freestyle	01:59.381	854	2023-05-05
9	800m freestyle	08:00.334	941	2022-11-05
23	200m breaststroke	02:27.746	852	2022-06-24
25	50m butterfly	00:26.955	826	2022-11-05
31	200m individual medley	02:10.529	873	2022-11-05

Slika 5.6: Kartica s prikazom detalja o plivaču

5.1.4. Bazeni

Odabirom kartice bazeni prikazuje se popis registriranih bazena. Pritiskom na strelicu na kartici bazena prikazuju se detalji o pojedinom bazenu (slika 5.7).



Slika 5.7: Kartica s prikazom svih bazena

5.1.5. Svjetski rekordi

Odabirom kartice Svjetski rekordi prikazuje se popis svjetskih rekorda u pojedinoj disciplini (slika 5.8).

World Records						
Name	Surname	Country	Event	Gender	Time	Meet
César	Cielo	Brazil	50m freestyle	M	00:00:20.910	World Championship Rome 2009
David	Popovici	Romania	100m freestyle	M	00:00:46.860	World Championship Rome 2022
Paul	Biedermann	Germany	200m freestyle	M	00:01:42.000	World Championship Rome 2009
Paul	Biedermann	Germany	400m freestyle	M	00:03:40.070	World Championship Rome 2009
Zhang	Lin	China	800m freestyle	M	00:07:32.120	World Championship Rome 2009
Sun	Yang	China	1500m freestyle	M	00:14:31.020	Olympic Games London 2012
Hunter	Armstrong	United States	50m backstroke	M	00:00:23.710	2022 USA Swimming International Team Trials
Thomas	Cecon	Italy	100m backstroke	M	00:00:51.600	2022 World Aquatics Championships
Aaron	Peirsol	United States	200m backstroke	M	00:01:51.920	World Championship Rome 2009
Adam	Peaty	Great Britain	50m breaststroke	M	00:00:25.950	2017 World Aquatics Championships
Adam	Peaty	Great Britain	100m breaststroke	M	00:00:56.880	2019 World Aquatics Championships
Zac	Stubblety-Cook	Australia	200m breaststroke	M	00:02:05.950	World Championship Rome 2009
Andriy	Govorov	Ukraine	50m butterfly	M	00:00:22.270	World Championship Rome 2009
Caeleb	Dressel	United States	100m butterfly	M	00:00:49.450	Olympic Games Tokyo 2020
Kristóf	Milák	Hungary	200m butterfly	M	00:01:50.340	2022 World Aquatics Championships
Ryan	Lochte	United States	200m individual medley	M	00:01:54.000	2011 World Aquatics Championships
Michael	Phelps	United States	400m individual medley	M	00:04:03.840	Olympic Games Beijing 2008
					rows per page: 17	1-17 of 34 < 1 2 >

Slika 5.8: Kartica s prikazom svjetskih rekorda

6. Zaključak

Web aplikacija za praćenje plivačkih rezultata i pregledavanje podataka o plivačkim natjecanjima uspješno je ostvarena uporabom navedenih alata i tehnologija. Ova aplikacija je intuitivna za korištenjem sadašnjim, bivšim ili budućim plivačima, ali i ljubiteljima plivanja. Ostvareno je jednostavno korisničko sučelje za prikaz navedenih podataka.

Razvoj sustava podijelio sam u dvije faza: modeliranje i implementacija baze podataka, te pisanje koda za razvoj web aplikacije.

Aplikacije bi se mogla unaprijediti dodavanjem različitih korisnika i trenera koji bi mogli mijenjati i unositi podatke. Također bi se mogla dodati opcija usporedbe dvaju plivača na temelju njihovih rezultata ili graf prikaza napretka plivača kroz godine. Takve opcije bi omogućile plivačima da prate svoj napredak i imaju uvid u sve svoje rezultate.

Rad na projektu mi se svidio jer je ovo prva veća aplikacija koju sam samostalno izradio. Izrada se pokazala kao vrlo poučno iskustvo jer sam usavršio vec poznate tehnologije i naučio nove koje ću moći primjenjivati u budućim projektima. Smatram da je znanje stečeno u izradi praktičnog proizvoda korisnije od teorijskog znanja te se osjećam samopouzdanije u svoje programerske sposobnosti.

LITERATURA

FER: Baze podataka, Predavanja, 2022..

FER: Programsko inženjerstvo, Predavanja, 2022..

Draw.io alat za crtanje dijagrama obazaca uporabe. URL <https://app.diagrams.net/>.

Vuetify UI biblioteka. URL <https://vuetifyjs.com/en/>.

DBeaver user guide. URL <https://dbeaver.com/doc/dbeaver.pdf>.

PostgreSQL documentation. URL <https://www.postgresql.org/docs/>.

Java Spring tutorial. URL <https://www.javatpoint.com/spring-tutorial>.

MVC framework introduction - tutorialspoint. URL https://www.tutorialspoint.com/mvc_framework/mvc_framework_introduction.htm.

Vue.js 3 - chart components. URL <https://vue-chart-3.netlify.app/>.

POPIS SLIKA

2.1. Dijagram obazaca uporabe	3
3.1. ER model baze podataka	12
3.2. Relacijski model baze podataka	20
4.1. Razred Natjecanje - model	24
4.2. Sučelje NatjecanjeRepository - repozitorij	25
4.3. Razred NatjecanjeController - kontroler	25
4.4. Vue.js akcija za dohvata podataka sa servera	26
4.5. Vue.js komponenta za prikaza kartica natjecanja	26
5.1. Kartica s prikazom svih natjecanja	28
5.2. Kartica s prikazom detalja o odabranom natjecanju	28
5.3. Forma za dodavanje novog natjecanja	29
5.4. Kartica s prikazom kvalificiranih plivača i limita	29
5.5. Kartica za pretraživanje plivača	30
5.6. Kartica s prikazom detalja o plivaču	30
5.7. Kartica s prikazom svih bazena	31
5.8. Kartica s prikazom svjetskih rekorda	32

Baza podataka i web-aplikacija za plivačka natjecanja

Sažetak

Baza podataka i web-aplikacija za plivačka natjecanja služi za pohranu plivačkih rezultata i podataka o plivačima. U uvodu je opisana glavna motivacija za izradu ovakve aplikacije. Kroz specifikaciju zahtjeva i opis obrazaca uporabe opisane su glavne značajke sustava, i koje korisničke zahtjeve sustav mora ispuniti.

Oblikovanje sustava odvojeno je u dva dijela: izrada baze podataka i izrada web-aplikacije. Prije početka pisanja koda bilo je potrebno zamisliti što je sve potrebno u bazi podataka, te misli preusmjeriti na ER model i relacijski model. Koristeći PostgreSQL sustav za baze podataka, modeli su pretvoreni u realizaciju. Drugi dio oblikovanja sustava sastojao se od oblikovanja poslužiteljske strane u Java Springu, koja izravno komunicira s bazom, te klijentske strane, koja predstavlja korisničko sučelje.

Ključne riječi: Plivanje, specifikacija zahtjeva, PostgreSQL, baza podataka, Java Spring, klijent, poslužitelj

Database and web application for swimming competitions

Abstract

Database and web application for swimming competitions is used for storing swimming results and data about swimmers. The main motivation for implementing such an application. Throughout use of system requirements and use case description the main, features of the system are outlined, as well as user requirements which system must meet.

System design is separated in two parts: database implementation and web application development. Before starting the coding process, it was necessary to think about database requirements and then use that requirements to model ER model and relational model. Using PostgreSQL system for databases, models were made into existence. Second part of implementation involved using Java Spring to shape the server side, which communicated directly with the database, and client side, which represents user interface.

Keywords: Swimming, case specification, PostgreSQL, database, Java Spring, client, server