

# SofaScore Edukacija

## 8. Domaća zadaća

### Priprema za završni projekt: model podataka

Alen Murtić, alen.murtic@sofascore.com  
Karlo Knežević, knezevic.karlo1@gmail.com

23. svibnja 2021. - 3. lipnja 2021. u 8:00

## 1 Uvod

Završni projekt SofaScore Edukacije prikazat će razumijevanje gradiva svih predavanja te upoznaje polaznike sa SofaScore domenom. Projektni zadatak bit će izrada backenda za unos i izmjenu podataka jedne aplikacije sportskih rezultata. Za to to potreban ispravan model podataka. Zato je zadatak 8. domaće zadaće upravo to - izrada modela podataka na koji će biti dodana aplikacijska logika završnog projekta.

Ispravan model podataka nosi 30 bodova u završnom projektu, što znači da će svaki polaznik akademije trebati skupiti samo dodatnih 20 bodova za minimalan prolazak završnog projekta.

### 1.1 Model podataka



Slug je transformacija stringa tako da može uključiti u simbolički URL bez enkodiranja. Nije nužno jednoznačan, npr. Đoković se može pretvoriti u Djokovic ili Dokovic, to su implementacijski detalji programske knjižnice koja se koristi (iako kao poznavatelji hrvatskog jezika znamo da je djokovic preciznije). U sklopu ovog projekta pretpostavit ćemo da slug sadrži samo mala slova i znak minus. Npr. očekivani slug stringa "Marin Čilić" će biti "marin-cilic". Da bi se taj string mogao uključiti u URL bez transformacije u slug, bilo bi ga potrebno enkodirati PHP funkcijom `urlencode` čiji bi rezultat bio sasvim nešto drugo. U ovom projektu slugifikacija ne mora biti jako pametna, važno je da je implementirana. Možete koristiti neku vanjsku knjižnicu koja to radi, sami napisati nešto ili iskoristiti rješenje s interneta. U slučaju da je slugifikacija dio projekta (znači druga ili treća opcija iz prošle rečenice), potrebno je napisati testove za nju.

Potrebno je podržati dva sporta - nogomet i košarku.

### 1.1.1 Sport

Sport je osnovni razred modela, a sadrži sljedeće podatke:

- identifikator (int)
- ime (string) - npr. Football, Basketball, itd.
- slug (string) - npr. football, basketball, itd.

Slug je **jedinstven samo nad sportom** i samo za sport se koristi umjesto identifikatora u rutama, tj. ruta koja prima sport parametar neće biti izvedena kao /sport/id (npr. sport/1), nego /sport/slug (npr. /sport/football ili samo /football).

### 1.1.2 Category (kategorija)

Category je entitet koji modelira prostornu ili organizacijsku značajku pojedinog natjecanja. Njezina funkcionalnost unutar projekta je samo semantičko grupiranje natjecanja. Primjeri kategorija su: Hrvatska, Engleska, Europa, Formula 1, WTA, ATP, ITF, Davis Cup. Važno je iz modela ispod uočiti da je kategorija vezana uz sport pa tako postoji kategorija Hrvatska za nogomet, Hrvatska za košarku, Hrvatska za rukomet i slično. Model Category treba izgledati ovako:

- identifikator (int)
- ime (string) - npr. Croatia, England, WTA, ATP
- slug (string)
- sport (Sport) - sport kojoj kategorija pripada

### 1.1.3 Competitor (natjecatelj)

Competitor je entitet koji se natječe protiv drugih Competitora. U nogometu, košarci rukometu i sličnim sportovima to je klub ili reprezentacija, tj. momčad. U sportovima poput tenisa, pikada ili stolnog tenisa, competitor je osoba. Nadalje, competitor može biti više od jedne osobe koji ipak nisu klub, npr. parovi u tenisu, stolnom tenisu ili badmintonu.

Da bi se osigurala jednostavnost implementacije Match modela, **svi tipovi natjecatelja se spremaju u jednu tablicu, competitor**. Ta tablica mora sadržavati atribute dovoljne za opis svih navedenih tipova competitora. Oni su:

- identifikator (int)
- ime (string) - ime kluba ili osobe, za parove treba biti konkatencija imena osoba odvojenih crticom
- slug (string)

- sport (Sport) - sport natjecatelja
- država (Country) - država kluba ili reprezentacije, državljanstvo osobe ili parova (ako oba imaju isto državljanstvo). Iskoristiti embedded entitete. Country nad natjecateljem nije nullabilan, ali kod države unutar Country objekta je.
- type (int ili string) - određuje je li competitor, tim/momčad, osoba ili par (npr. parovi u tenisu). U slučaju da je competitor modeliran apstraktnim razredom Competitor i trima konkretnim razredima (Team, Person i Pair) u obliku single table nasljeđivanja, tada je type string. Ako postoji samo konkretan razred Competitor, tada je type int.

#### 1.1.4 Competition (natjecanje)

Competitori se jedni protiv drugih natječu u sklopu nekog natjecanja - Competitiona. Svako natjecanje pripada jednoj kategoriji, npr. UEFA Champions League pripada kategoriji Europe, 1. Hrvatska nogometna liga pripada kategoriji Croatia. Natjecanje na sebi nema sport jer ga ima kategorija. Dakle, model natjecanja je:

- identifikator (int)
- ime (string)
- slug (string) - transformacija imena tako da se ono može uključiti u simbolički url
- kategorija (Category)
- koliko puta natjecatelji igraju jedni protiv drugih u sezoni (int)

U sklopu ovog projekta možete potpuno ignorirati natjecanja od više faza, više grupa, ona koja imaju sustav na ispadanje, doigravanja i isključivo razmatrati ligaška natjecanja poput engleske Premier lige ili HNL-a (bez doigravanja za ispadanje). Zato je ključno imati informaciju koliko puta timovi igraju međusobno u sezoni (npr. za Premier ligu je to 2, a za HNL 4).

*Napomena: Alternativni naziv za Competition je Tournament. Iako mislimo da je Competition sa jezičnog aspekta ispravniji naziv, u projektu možete koristiti Tournament iz banalnog razloga da kad krenete pisati "Comp" u PHPStormu, on ponudi samo Competitor razred, a ne i Competition te si time ubrzate pisanje koda.*

*Ako se pitate što mi u SofaScoreu koristimo - kod nas su svi Competitori u klasi Team (da, Roger Federer je momčad), faze Competitiona (npr. regularna sezona u NBA, grupa u Ligi Prvaka) su nam Tournament, a NBA i Liga Prvaka UniqueTournament. Zašto tako čudno? Takvo imenovanje je davno preuzeto od vanjskih suradnika i sad je potpuno neisplativo refaktorirati.*

### 1.1.5 Season (sezona)

- identifikator (int)
- ime (string) - npr. "2018-19" ili "2020"
- opcionalno: datum početka (datetime) - datum i vrijeme početka prve utakmice u sezoni
- opcionalno: datum završetka (datetime) - datum i vrijeme početka zadnje utakmice u sezoni

Prilikom implementacije slobodni ste odabrati postoji li u sustavu samo jedna "2019-20" sezona ili svako natjecanje koje traje od datuma u 2019. do datuma u 2020. ima svoju "2019-20" sezonu. Ako odaberete prvu opciju, treba napraviti entitet **CompetitionSeason** koji mora imati datum početka i datum završetka sezone, a za drugu opciju opcionalni atributi sezone postaju obvezujući. Dakle, oni moraju negdje biti zapisani, samo je pitanje gdje. U tom slučaju, sezona mora imati i referencu na Competition.

### 1.1.6 Match (utakmica)

Utakmica, dvoboj natjecatelja unutar natjecanja u nekoj sezoni. Postoji jedan apstraktan razred Match koju nasljeđuju izvedenice za nogomet i košarku. Dosašnji model je bio nezavisan o sportu, ali Match ne može biti jer se nogomet igra u dva poluvremena, a košarka u četvrtinama. Slobodno možete zanemariti košarku koja se igra u poluvremenima (NCAA). Model Matcha je:

- identifikator (int)
- domaći natjecatelj (Competitor)
- gostujući natjecatelj (Competitor)
- početak (datetime)
- status (int, konstante u kodu) - opcije: nije počela, 1. poluvrijeme, 2. poluvrijeme, pauza, 1.-4. četvrtina, otkazana, završena, produžeci (košarka)
- natjecanje (Competition)
- sezona (Season)
- domaći rezultati po periodima (Score)
- gostujući rezultati po periodima (Score)
- kod pobjednika (int) - null ako utakmica nije završena, 1 ako je pobijedio domaći natjecatelj, 2 gostujući, 3 neodlučeno

Budući da ovaj projekt pokriva samo ligaški sustav natjecanja, nogomet nema produžetaka.

Možete primijetiti da se neka natjecanja igraju na neutralnom terenu te nemaju gostujućeg i domaćeg natjecatelja, ali to je iz perspektive nazivlja sretnija opcija nego natjecatelj A i B. Ako smislite bolje nazivlje, slobodno ga koristite.

#### 1.1.7 Score (raspisani rezultat)

Score je embedded entitet koji sadrži rezultate po periodima i konačni rezultat kao što je opisano u 10. predavanju.

Atribut Score razreda koji se nije dogodio (npr. svi prije početka utakmice ili rezultat produžetaka ako je utakmica završila u regularnom vremenu) **mora biti null**.

#### 1.1.8 Standings i StandingsRow, tablica i redak u tablici

Standings je tablica, poredak timova za natjecanje u sezoni. Osim natjecanja i sezone, sadrži retke, entitete StandingsRow za svaki od natjecatelja koji igra u sezoni. Sadrži njegov broj pobjeda, poraza i neodlučnih rezultata te broj bodova i postotak pobjeda. Standings može biti ukupni (total), domaći i gostujući. Dakle, svako natjecanje u sezoni ima tri različita standingsa.

**VAŽNA NAPOMENA:** Čvrsto preporučamo da Standingsi budu implementirani jednim razredom Standings koji ima string "type" za razlikovanje jesu domaći, gostujući ili ukupni. I "single table", i "class table" nasljeđivanje bi učinilo korištenje standingsa značajno kompliciranijim.

Budući da projekt obrađuje samo ligaški sustav, retke standingsa koristite i kao popis natjecatelja u sezoni natjecanja.

Ako se odlučite za model u kojem sezona ima Competition, startDate i endDate, onda standings ne mora imati natjecanje jer to određuje sezona.

Dakle, mogući model standingsa je:

- identifikator (int)
- natjecanje (Competition)
- sezona (Season)
- type (string) - vrijednosti: home, away, total

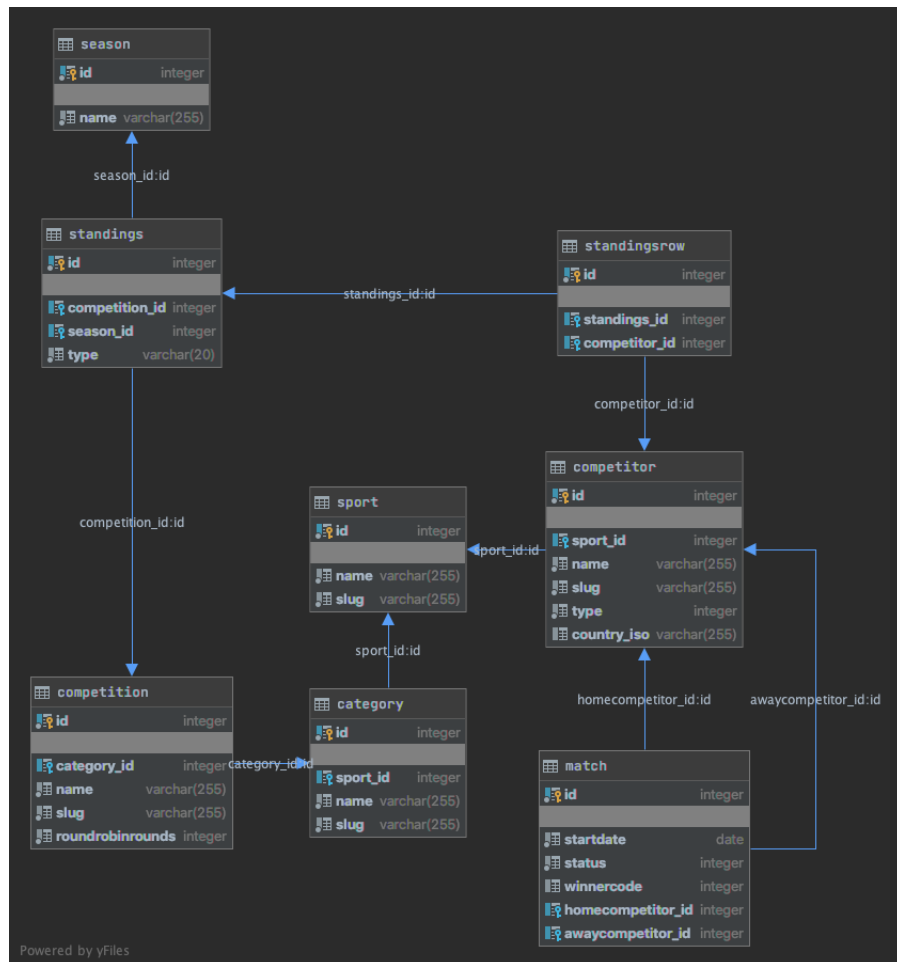
A retka u standingsima:

- identifikator (int)
- natjecatelj (Competitor)
- standings (Standings)
- matches (int) - broj završenih utakmica - incijalna vrijednost 0

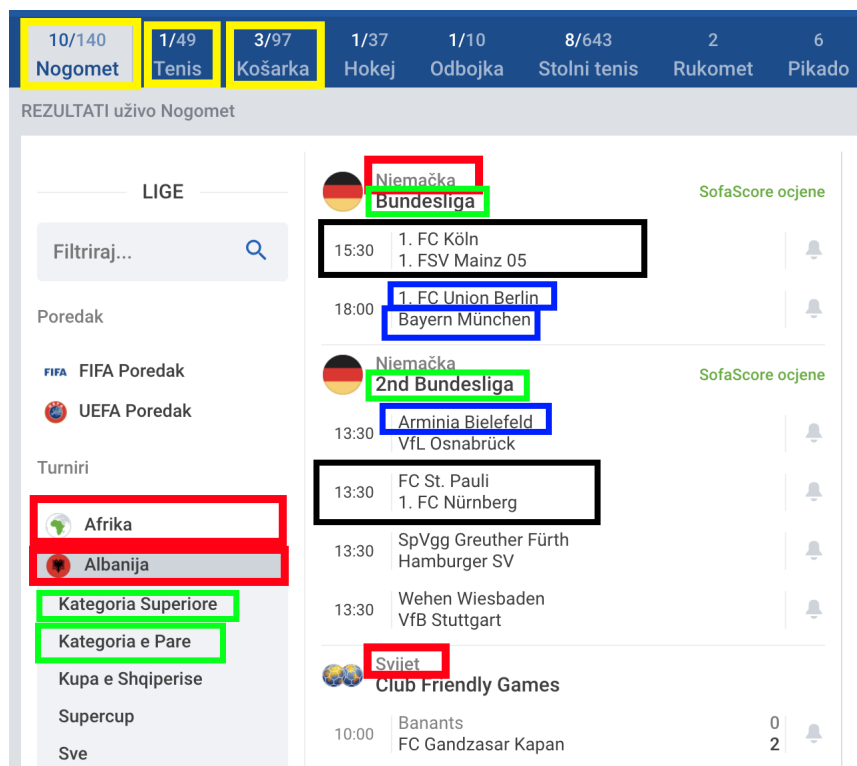
- wins, losses, scoresFor, scoresAgainst (int, inicijalna vrijednost im je 0)
- draws, points (nullabilni int) - ne postoje u košarci
- percentage (nullabilni float) - udio pobjeda u ukupnom broju utakmica, ne postoji u nogometu

Na slici 1 nalazi se predloženi E-R model podataka s izostavljenim stupcima nad Match i StandingsRow entitetima da bi bio lakše čitljiv.

Slike 2 i 3 prikazuju označene entitete na SofaScore stranici. Entiteti su označeni bojama: žuta - Sport, crvena - Category, crna - Match, tamno plava - Competitor, zelena - Competition, svijetlo plava - Standings, ružičasta - StandingsRow, narančasta - Season.




Slika 1: Predloženi E-R model podataka




Slika 2: Stranica sporta na SofaScore web stranici

## 2 Literatura

Za uspješno rješavanje ove domaće zadaće potrebno je razumijevanje predavanja vezanih uz Symfony i Doctrine. Dodatno, Symfonyjeva službena dokumentacija je izvrstan resurs jednostavnih programskih primjera i objašnjenja. Ona se nalazi na poveznici: <https://symfony.com/doc/current/index.html>.



**1. HNL**

 Croatia
 19/20











Receive notifications for all games in this tournament
 

FOLLOWING
 18k followers

20 JUL
 2 JUN

STANDINGS

OVERALL HOME AWAY

		P	W	D	L	Goals	Last 5	PTS
1	 Dinamo	26	21	2	3	51:11	L W W W W	65
2	 Rijeka	26	14	5	7	42:31	D W L W W	47
3	 Lokomotiva	26	14	4	8	41:28	W W W W W	46
4	 Hajduk	26	13	6	7	41:23	L W D L L	45
5	 Osijek	26	11	9	6	38:25	W L D L W	42
6	 Gorica	26	9	8	9	32:40	D L W D L	35
7	 Slaven	26	7	5	14	23:43	W D D L W	26
8	 Istra	26	4	7	15	19:40	L D L W L	19
9	 Inter	26	3	8	15	28:51	L L L L L	17
10	 Varaždin	26	3	8	15	18:41	W L D D L	17


In the event that two (or more) teams have an equal ...
 More



[Add live standings to your website!](#)

MATCHES

BY DATE BY ROUND

< Previous
 Next >

1. HNL
 02/05/20
 Postponed
 NK Osijek
 NK Istra 1961
 

Croatia > 1. HNL, Round 36
 NK Inter Zaprešić - HNK Hajduk Split
 
 -
 

Slika 3: Stranica natjecanja na SofaScore web stranici