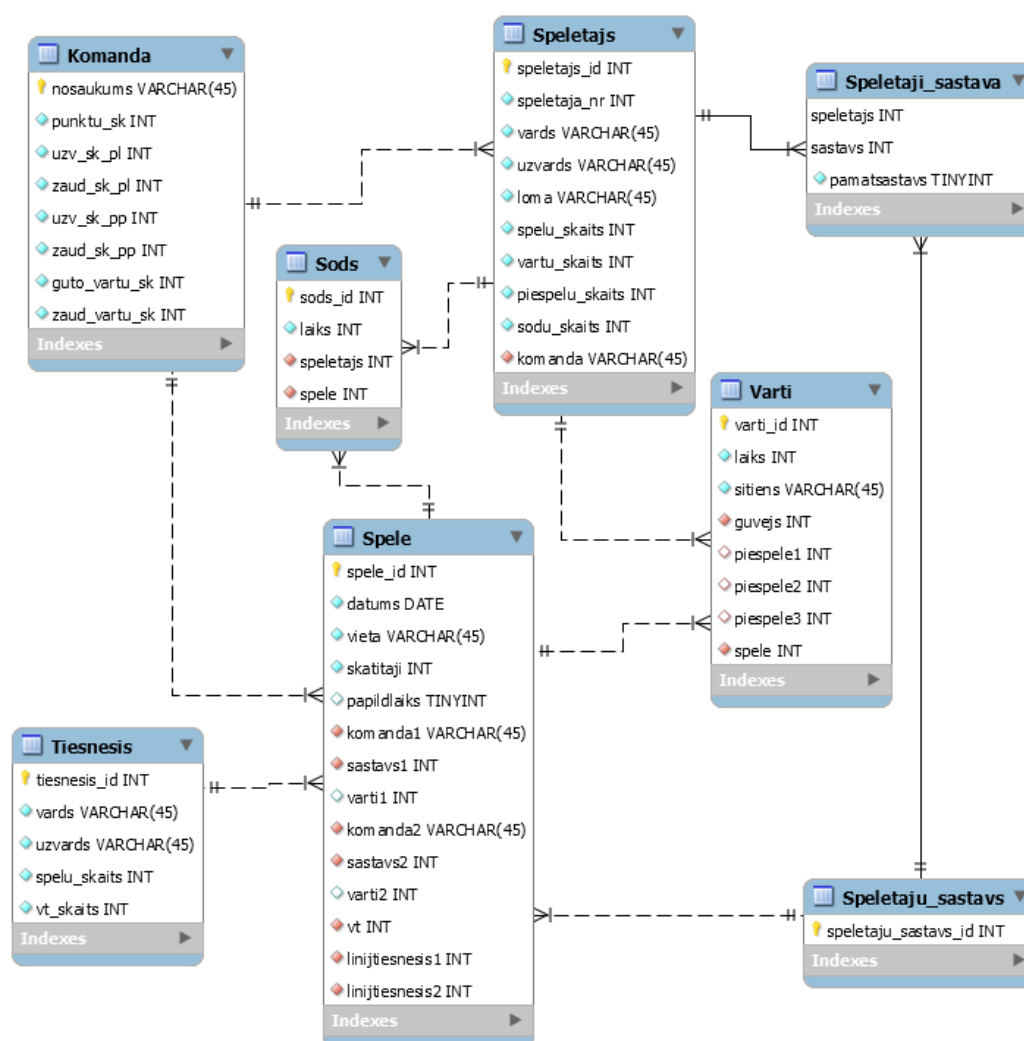


PD2 risinājuma apraksts

Uzdevuma atrisināšanai tika izvēlēta programmēšanas valoda **Python**, galvenokārt, tāpēc, ka autoram jau ir iepriekšēja pieredze strādājot ar šo programmēšanas valodu, kā arī uzstādīt un izmēģināt dažādas bibliotēkas, kuras var būt noderīgas projektam ir viegli, izmantojot **pip** pakotņu pārvaldnieku. Izmantotās pakotnes, kas nav **Python** sākotnējā uzstādījumā: **pysqlite3** un **plotly**.

Risinājumā tika izvēlēts veidot relāciju datu bāzi, kurā saglabāt informāciju par turnīru, lai pēc tam no tās ērti izņemt arī interesējošos datus. Modeļa izveidei tika izmantots rīks **MySQL Workbench** (skat. 1. att.)



1. Attēls – izveidotās datu bāzes modelis.

Modelis sastāv no 8 tabulām. Tabula “Komanda” sastāv pamatinformāciju par turnīra komandām. Tā ir saistīta ar tabulu “Spēlētājs”, kurā norādīts spēlētāja vārds, uzvārds, numurs un citi statistikas rādītāji. Tabula “Tiesnesis” satur informāciju galveno informāciju par tiesnešiem.

Tabula “Spēle” apraksta kādu no notikušajām spēlēm – tā satur pamatinformāciju kā datumu, vietu un rezultātu. Tā ir saistīta ar tabulām: “Tiesnesis” (spēlē ir viens virstiesnesis un divi līnijtiesneši) un “Komanda” (katrā spēlē ir divas komandas). “Vārti” ir tabula, kas apraksta vārtu guvumus un sasaista “Spēles” tabulu ar “Spēlētāju” tabulu. Līdzīgi tabula “Sods” apraksta sodus, kurus nopelnījis konkrēts spēlētājs spēlē.

Tabula “Spēlētāju_sastāvs” apraksta komandas sastāvus, kas piedalījās spēlē (katrai komandai ir savs sastāvs) un tabula “Spēlētāji_sastāvā” nosaka kādi spēlētāji pieder noteiktajam sastāvam.

Svarīgi pieminēt, ka datu bāze nav normalizēta un tas ir apzināti – komandas uzvaru skaitu pamatlaikā varētu aprēķināt tikai no tabulas “Spēle” ierakstiem, taču tas prasītu papildus sarežģītus vaicājumus un tika pieņemts lēmums uzkrāt statistikas datus katrā tabulā, lai tos būtu vieglāk atlasīt.

Pēc modeļa izveides, tika uzģenerēts SQL kods, datu bāzes izveidei, taču sakarā ar dažādām nesaderībām, to bija nepieciešams izmainīt, lai to būtu iespējams ielasīt ar izmantoto *pysqlite3* pakotni.

Risinājums sastāv no trīs posmiem – datu bāzes modeļa izveides, datu ievietošanas datu bāzē un statistikas izveides. Katram posmam tika izveidots savs **Python** skripta fails. Skriptā “*generate_database.py*” tiek izveidots jauns datu bāzes fails “*futbols.db*” un tiek atvērts izveidotais SQL skripta fails “*database_model.sql*”, lai izveidotu datu bāzes tabulas.

Skripts “*parse_data.py*” satur datu ievietošanas loģiku. Tiek ielasīti visi XML faili zem norādītās direktorijas (tiek nolasītas arī apakšdirektorijas un tajos esošie XML faili) un to nosaukumi tiek saglabāti sarakstā. Iteratīvi tiek ielasīts katrs fails un tiek nolasīta pamatinformācija par spēli kā datums un vieta. Pēc tam pārbauda, vai komandas, kuras piedalās spēle jau ir datu bāzē, ja nav, tad tās ievieto. Šajā posmā arī tiek pārbaudīts, vai komanda jau nav šajā dienā spēlējusi. Pēc komandu ielases, tiek ievietoti visi spēlētāji un tiesneši, kas nav datu bāzē. Pēc tam – sodi un vārtu guvumi tiek ievietoti datu bāzē. Svarīgi pieminēt, ka arī

tabulās “Spēlētājs” tiek palielināti ieraksti kā “vārtu_skaits”, “piespēlu_skaits”, “sodu_skaits” un “spēlu_skaits”. Kad nolasīta šī informācija, tiek noteikts spēles uzvarētājs un tabulā “Spēle” tiek norādīta trūkstoša informācija kā 1. un 2. komandas guvušo vārtu skaits. Izmaina arī tabulas “Komanda” uzvaru un zaudējumu skaitu pamatlaikā vai papildlaikā, kā arī tiek pieskaitīts attiecīgo punktu skaits.

Kad visa informācija ir ievietota datu bāzē, tad iespējams uzģenerēt statistiku. Skriptā “*generate_statistics.py*” tiek izveidotas 4 tabulas, kas atbilst 4 statistikas rādītājiem:

1. Turnīra tabula
2. Turnīra 10 rezultatīvākie spēlētāji
3. Spēles ar visvairāk gūtajiem vārtiem
4. Pieredzējušākie tiesneši (ar visvairāk notiesātajām spēlēm)

Tabulu izveidei tika izmantota bibliotēka *plotly*, kas atļauj uzģenerēt tabulas HTML failos, norādot sarakstus ar elementiem, kurus nepieciešams ievietot tabulu kolonnās. Ar bibliotēkas palīdzību, no datu bāzes tiek izņemta nepieciešamā informācija, ievietota sarakstos un attēlota tabulās. Rezultātā tiek uzģenerēts fails “*statistics.html*”, kuru iespējams atvērt ar pārlūkprogrammu.