|  |
| --- |
|  |

**ANALIZA**

|  |  |
| --- | --- |
|  | Definicija fizičke arhitekture |
|  | AD.00.02 |

**WEB APLIKACIJA ZA**

**PRODAJU KARATA**

|  |  |
| --- | --- |
| Oznaka projekta | **PROJEKT PIS01** |
| Nositelj projekta | **TVZ** |
| Suradnici | **Nastavnici, studenti** |
| Korisnik projekta | **Referada, Studenti, Uprava, Nastavnici** |
| Datum dokumenta | **12.03.2022.** |
| Tekuća inačica | **0.1** |

# Nadzor dokumenta

**Promjene dokumenta**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Inačica | Autor | Tag | Datum | Opis |
| 0.1 | Nikola Platnjak | #MO | 27.3.2022 | Izrada dokumenta i popunjavanje |
| 0.2 | Dorian Hajnić | DH01 | 13.3.2024. | Opis arhitekture i komunikacije |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Revizori/Nadzor**

|  |  |  |
| --- | --- | --- |
| Ime i prezime | Pozicija | Napomena |
| Nikola Platnjak | Voditelj tima |  |
| Dorian Hajnić | Razvoj softvera |  |
| Kristina Aničić | Razvoj softvera |  |
| Nina Šalković | Testiranje |  |

**Raspodjela dokumenta**

|  |  |  |
| --- | --- | --- |
| Br. kopije | Ime i prezime | Lokacija |
| 1 | Nikola Platnjak | GitHub |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Sadržaj

[1 Nadzor dokumenta 2](#_Toc99304838)

[2 Sadržaj 3](#_Toc99304839)

[3 Opis arhitektura 4](#_Toc99304840)

[4 Opis komunikacije 5](#_Toc99304841)

[5 Grafički prikaz 5](#_Toc99304842)

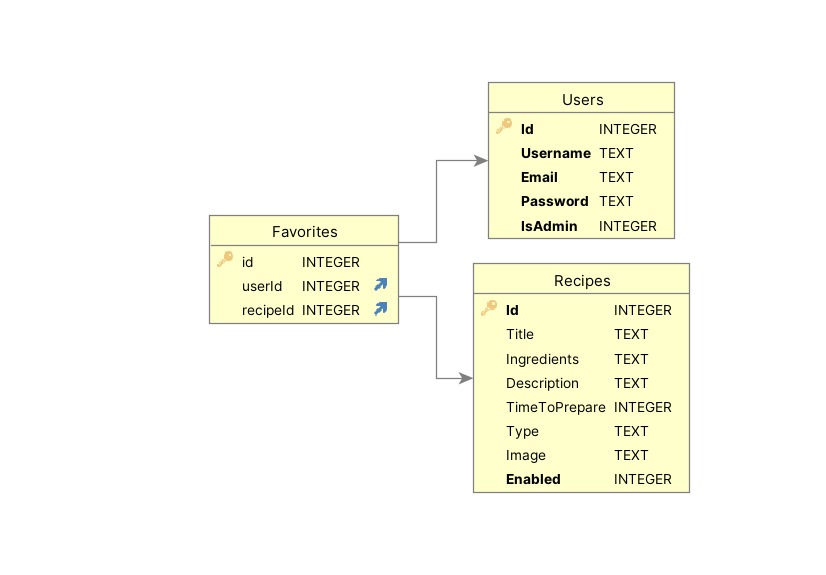
# Opis arhitektura

**#DH01**

Web aplikacija će se sastojati od tri komponente. Frontend dio biti će razvijen u Angular-u. Za Angular smo se odlučili zbog prijašnjeg iskustva programera. Zbog malog broja podataka baza koju smo odlučili koristiti je SQLite. Baza će se sastojati od 3 tablice koje će sadržati recepte, korisnike i favorite. Za backend dio aplikacije odabrali smo ASP.NET Core u jeziku C#. Razlog za taj odabir je prijašnje iskustvo s jezikom C# i želja programera da nauči ASP.NET. Uz to koristiti ćemo Entity Framework Core. Entity Framework Core je open-source ORM framework. ORM framework je tehnologija koja automatizira pretvaranje podataka između objekata i baze podataka.

Aplikacija je razvijena po MVC (Model-View-Controller) obrazcu. MVC se koristi u softverskom inženjeringu za odvajanje pojedinih dijelova aplikacije u komponente ovisno o njihovoj namjeni.

**#DH01**



Slika 1 - Struktura baze podataka

# Opis komunikacije

**#DH01**

Komunikacija počinje od klijentovog preglednika koji šalje zahtjev serveru. API u serveru procesira zahtjev, pristupa bazi podataka i generira odgovor. API se sastoji od klasa modela, repozitorija i kontrolera. Klase modela definiraju strukturu podataka. Klase repozitorija definiraju kako se podatci obrađuju. Klase kontrolera pozivaju metode klasa repozitorija s obzirom na zahtjev i generiraju odgovor. Frontend dio servera obrađuje odgovor te generira prikaz koji šalje korisniku.

**#DH01**

# Grafički prikaz

