

Sa FTP servera (folder **Download/RSII**, user/pass: **student**) preuzmite template projekta na kome ćete raditi implementaciju funkcionalnosti opisanih u nastavku. Promijenite konekcijski string kako biste se povezali na SQL server na kojem ćete razvijati i testirati svoj projekt.

- **Adresa i naziv servera:** 192.168.1.15\Ispiti
- **Port:** 1434
- **Baza podataka:** Koristeći migracije kreirajte bazu podataka sa vašim brojem indeksa npr. IB150051
- **Pristupni podaci:**

Username = exam

Password = exam1434

Svi studenti ispit rade na **istoj instanci servera**, pa je važno da radite isključivo unutar baze koja pripada vašem projektu.

1. Dodajte inicijalnu migraciju (add-migration) i kreirajte vašu bazu (update-database). Nakon uspješne migracije trebalo bi da postoje pristupni podaci *username=exam password=exam1434*.
Prilikom implementacije zadataka opisanih u nastavku neophodno je dosljedno poštovati postojeću arhitekturu i projektne principe. Implementacija treba uključivati dodavanje potrebnih (dto)entiteta, servisa i kontrolera unutar odgovarajućih slojeva, uz obavezno korištenje postojećih helper klasa i drugih već implementiranih mehanizama.
2. U okviru templejta projekta, na Web API dijelu, dodati entitet **FavoriteBrojIndeksa** koji služi za čuvanje liste omiljenih proizvoda svakog korisnika. Entitet treba sadržavati vezu na korisnika i proizvod, datum i vrijeme dodavanja proizvoda u listu omiljenih, te druge atribute koje budete smatrati potrebnim.
3. Na formi za prikaz svih proizvoda (*product_list.dart*) implementirati mogućnost da se klikom na neki od proizvoda pokreće nova forma sa detaljima proizvoda (npr. *product_details.dart*). Na toj formi potrebno je prikazati sve dostupne podatke o proizvodu, uključujući sliku, naziv, cijenu, opis i druge vrijednosti iz referentnih tabela prikazane kroz njihove nazine umjesto Id-eva (npr. jedinica mjere, tip proizvoda). Ne koristiti postojeće widget-e namijenjene dodavanju ili editovanju podataka već kreirajte nove namijenjene samo prikazu.

U okviru forme *product_details*, pored ranije opisanih detalja, prikazati dugmić ili ikonicu *Dodaj u omiljene* koji će omogućiti da se posmatrani proizvod doda u listu omiljenih, dok se ponovnim klikom isti proizvod uklanja iz liste omiljenih. Na odgovarajući način naznačiti da li se proizvod već nalazi u listi omiljenih.

4. Kreirati posebnu formu (npr. *favorite_list.dart*) za prikaz svih omiljenih proizvoda korisnika koja će posjedovati mogućnost njihovog filtriranja po vremenu dodavanja (OD - DO).
5. Pored osnovnih informacija o omiljenim proizvodima, na formi *favorite_list.dart*, dodati dugmić koji omogućava dodavanje svih omiljenih proizvoda direktno u Cart. U slučaju da postoji, omiljene proizvode dodati u Cart ili u suprotnom kreirati novi Cart (vodeći računa

o dodavanju potrebnih zapisa u CartItems). Nakon uspješnog dodavanja proizvoda u Cart, lista omiljenih proizvoda se prazni/briše za posmatranog korisnika.

6. Dodajte par testnih podataka (seed) za novokreirane entitete.
7. Na kraju ispita, u API projektu, obrišite bin i obj foldere (opcija Clean Solution) u okviru svakog projekta (Model, Services i API), a unutar Flutter projekta u terminalu izvršite komandu `flutter clean`.

Nakon toga, zapakujte projekat u folder imenovan vašim brojem indeksa i postavite na FTP server (folder **Upload/RSII**, user/pass: **student**)