

## Refactoring

### SoftwAIR

## OOAD

### Kopiranje koda

Na prvi problem prilikom implementacije smo se susreli kada smo zbog autentikacije morali ponovo praviti app.

Kopirali smo sve modele, kontrolere i viewe u novi, te je doslo do problema sa dll-ovima.

U konsultaciji sa drugim kolegama smo odlucili rucno opet napisati kod u novom projektu,

kako bi izbjegli sve potencijalne buduće probleme sa dll-ovima.

### Problemi sa abstraktnom klasom Korisnik

Naime, budući da smo imali samo klasu korisnik, a mnogo funkcionalnosti sistema nam se baziralo na odluci da li je korisnik registrovan ili ne,

zbog izgleda baze, a i funkcionalnosti sa Bridge paternom, odlucili smo se Korisnika napraviti abstraktnom klasom, iz koje slijede 2 klase : Registrovani i Neregistrovani Korisnik.

To nam je omogućilo lakše upravljanje bazom, te pravilniji slijed i postivanje pravila objektno orijentisanog dizajna.

### Migracija na bazu podataka

Najviše stvari sa kojim smo susreli problema jeste migracija baze.

Radili smo sa FreeSQL Database serverom.

Prvenstveno smo naisli na problem brisanja migracije, budući da nam je prva migracija

bila testna.

Nakon brisanja, imali smo problem sa objavljivanjem abstraktne klase na samu bazu.

To smo riješili tako što smo u AppDbContextu dodali diskriminator te tako abstraktnu klasu povezali sa naslijeđenim.

Naredni problem sa kojim smo se susreli jeste taj što nas je odluka da namjerno izostavimo "set" metode sa određenih klasa modela spriječavala da ih migriramo na bazu, stoga smo morali dodati "set" metodu za sve modele.

### Uređjivanje viewa i kontrolera prilikom scaffoldanja

Na poslijetku, bilo je potrebno dosta vremena da se očisti kod od mnogih (za nas specifičan projekat) "bespotrebnih" fileova koje generise scaffoldanje.