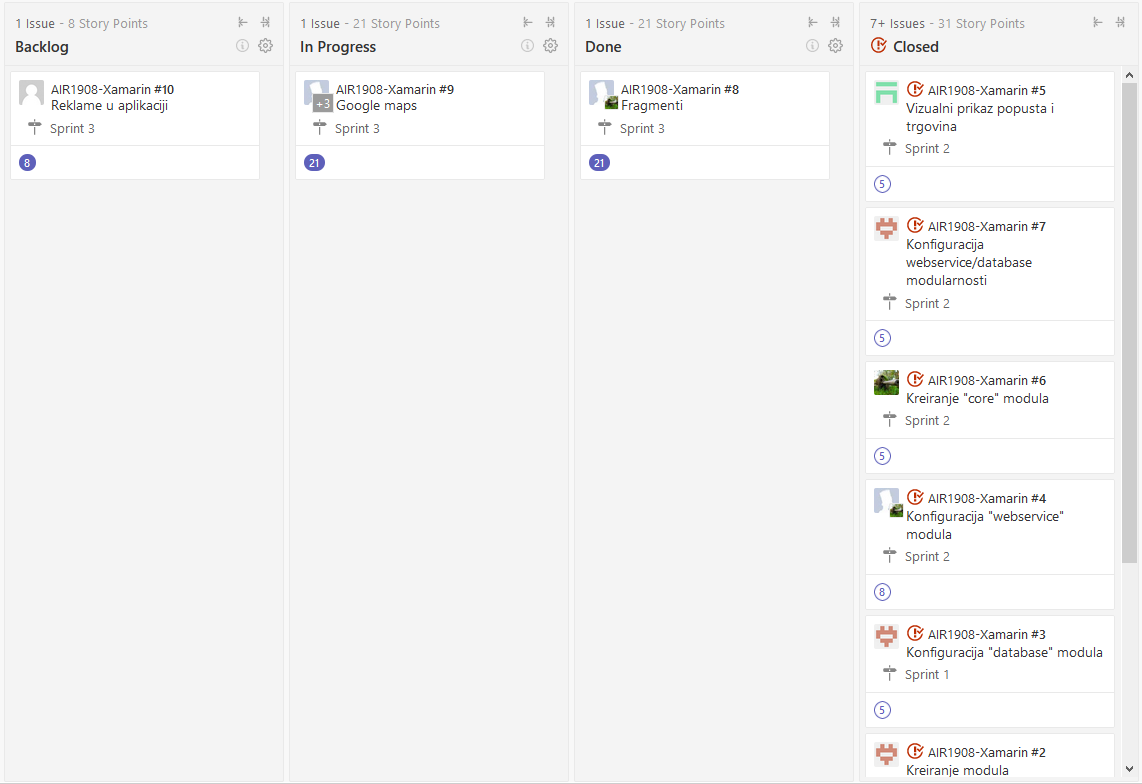
***Sažetak***

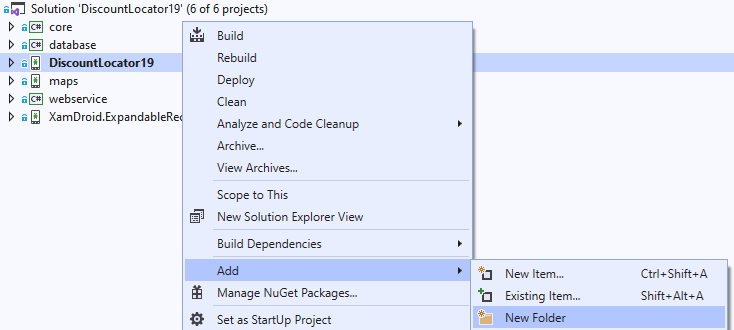
U prošloj vježbi pripremili smo mapu za prikaz dućana. U ovoj vježbi omogućit ćemo modularni prikaz podataka. Korisnik će u glavnom izborniku aplikacije odabrati kakav prikaz podataka želi (prikaz u listi ili na mapi).

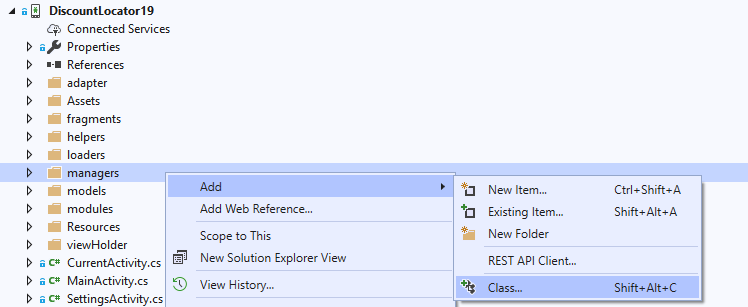
Zadnja verzija aplikacije: Commit 34fd9d3 [<https://github.com/AIR-FOI-HR/AIR1908-Xamarin>]

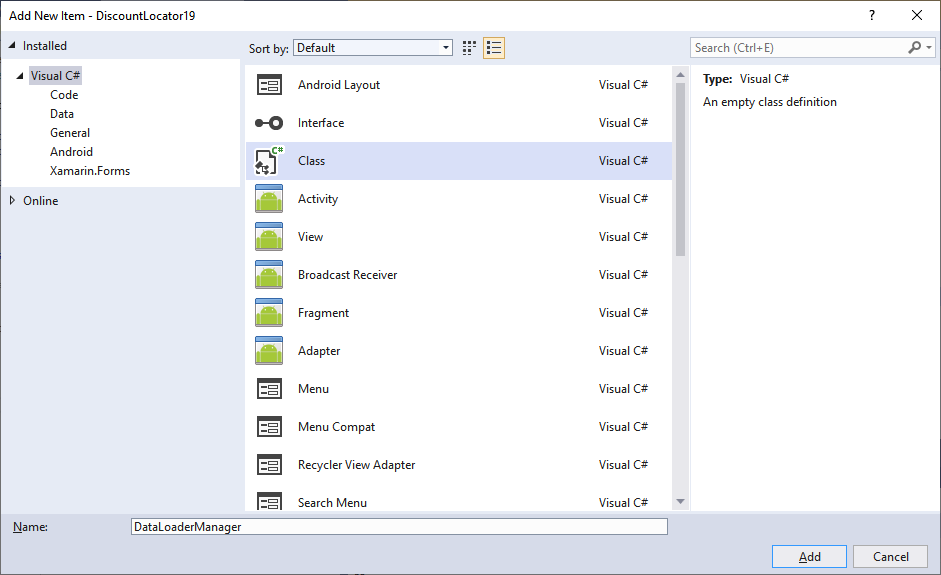
***Što radimo?***

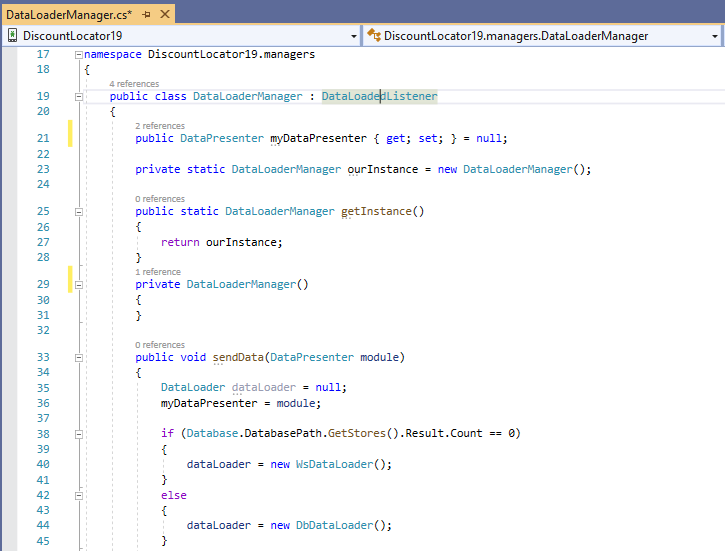


Za početak ćemo dodati novu klasu koja će biti zadužena za učitavanje podataka s web servisa i iz baze podataka. Nazvat ćemo ju DataLoaderManager i smjestit ćemo ju u novu mapu koju ćemo nazvati managers.



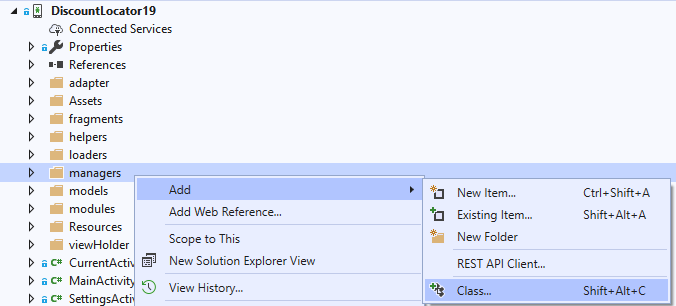


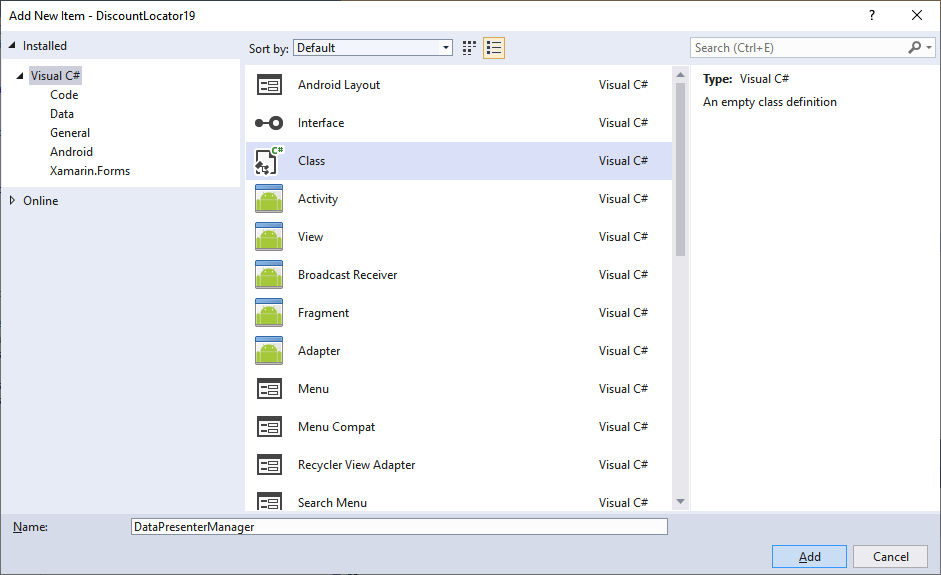




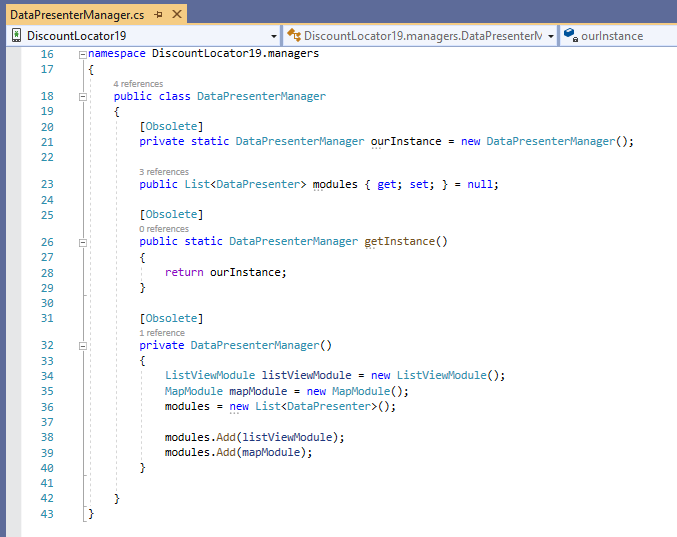
DataLoaderManager klasa nasljeđuje DataLoaderListener interface. Zato implementiramo metodu onDataLoaded. Stvorili smo jedan MyDataPresenter property tipa DataPresenter. Metoda sendData pozivat će se kad bilo koji odabrani modul zatraži podatke. U ovoj će se metodi odlučivati o izvoru podataka na isti način kako smo to već radili. Provjerit ćemo postoji li u bazi podataka spremljen dućan pa ćemo na temelju rezultata upita odlučivati o izvoru podataka.

Sada ćemo dodati novu klasu u mapu managers. Nazvat ćemo ju DataPresenterManager i bit će zadužena za dodavanje u listu svih modula zaduženih za prikaz podataka (to su ListViewModule i MapModule).

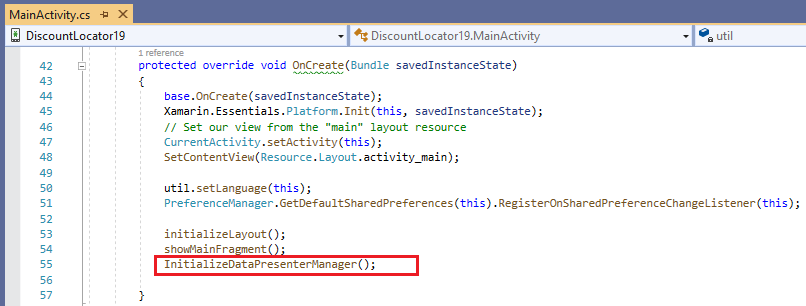


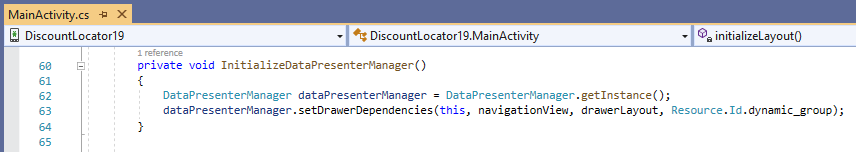


Dodat ćemo novu listu u koju ćemo spremati interface DataPresenter. U konstruktoru klase dodavat ćemo module u novokreiranu listu modules.

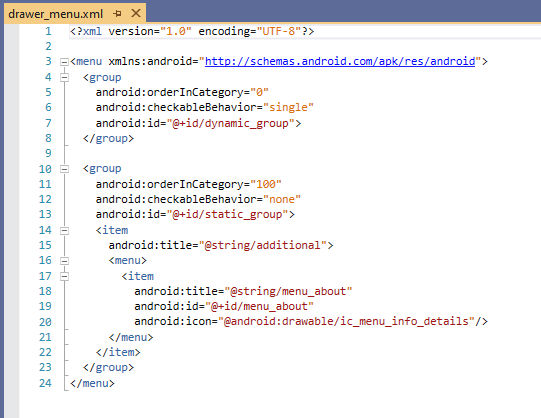


Sada ćemo se prebaciti na MainActivity. U OnCreate metodu dodat ćemo pozivanje InitializeDataPresenterManager() metode koja će pozivati metodu getInstance() DataPresenterManager klase. Tim se pozivom u listu modules dodaju dva modula koja ćemo koristiti za prikaz podataka.

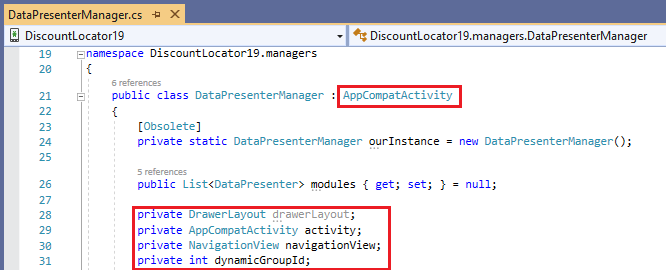




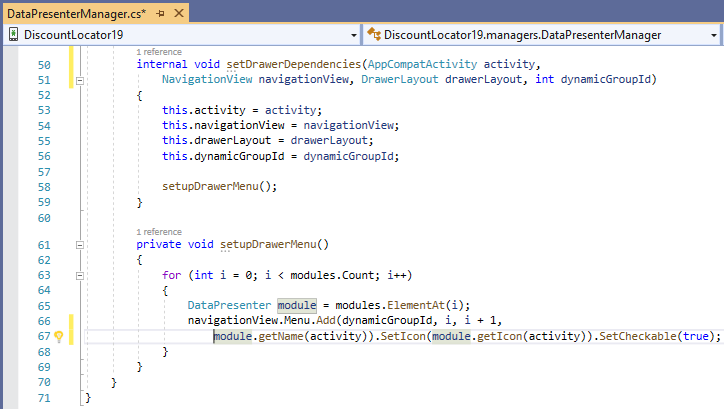
Potrebno je izmijeniti drawer\_menu.xml u menu mapi DiscountLocator19 modula.



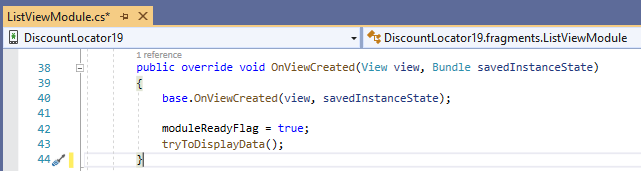
U DataPresenterManager klasi dodat ćemo nasljeđivanje AppCompatActivity klase. Dodat ćemo i nekoliko propertyja.



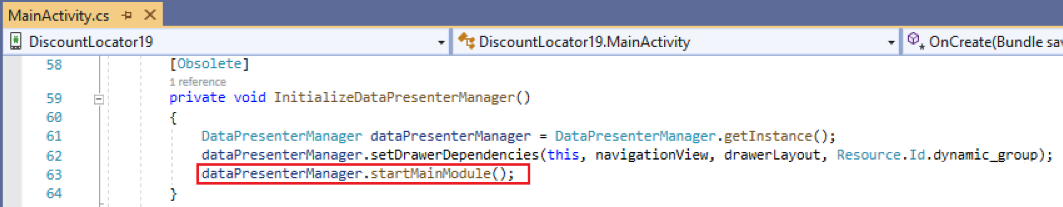
Implementirat ćemo dvije metode: setDrawerDependencies koju smo pozivali u InitializeDataPresenterManager() metodi u OnCreate() MainActivity klase i setupDrawerMenu() koja će dodavati elemente u glavni izbornik.



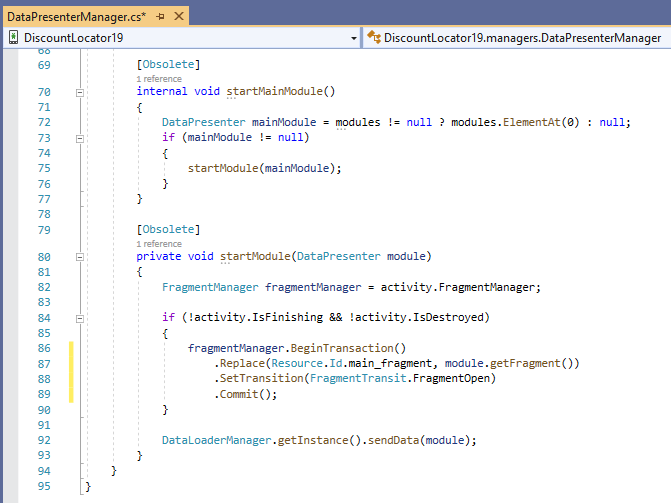
U ListViewModule klasi sada možemo izbrisati kod koji nam nije potreban. To je provjera stanja u bazi podataka i ispis poruke na ekran. Tako nam je u OnViewCreated() metodi potrebno samo sljedeće:



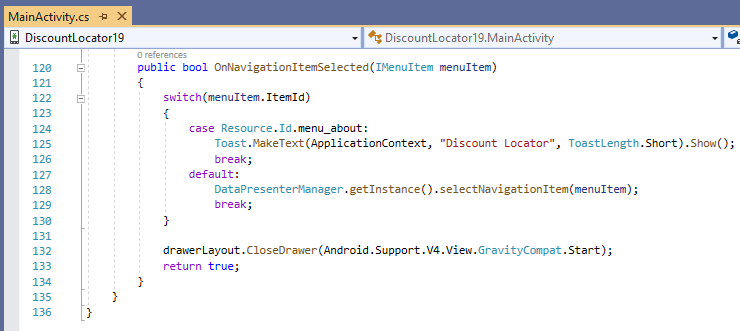
Sada ćemo u MainActivity klasi obrisati pozivanje metode showMainFragment(). Umjesto toga, dodat ćemo pozivanje metode startMainModule koju ćemo implementirati u DataPresenterManager klasi.



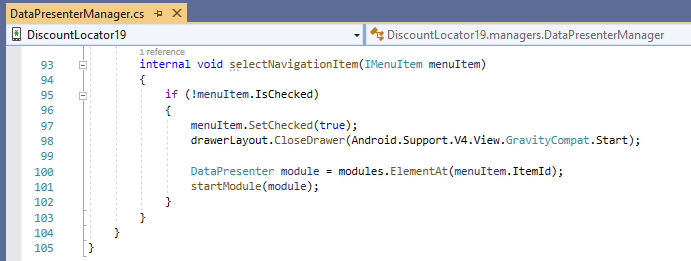
Ova će metoda pokretati ListViewModule pri pokretanju aplikacije.



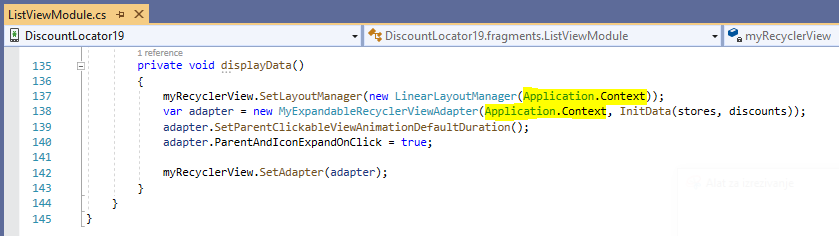
Sada ćemo omogućiti klik na stavke izbornika. U MainActivity klasi, u OnNavigationItemSelected() metodi. Ako korisnik odabere stavku 'O aplikaciji', prikazat će mu se poruka s nazivom aplikacije. Ako odabere bilo koju drugu stavku, pozvat će se getInstance() metoda DataPresenterManager klase, a zatim će se pozvati secetNavigationItem() metoda s identifikatorom odabrane stavke kao parametrom.



U selectNavigationItem() metodi odlučuje se koji će modul prikazivati podatke s obzirom na poslani identifikator. Modul se uzima iz liste modula i pokreće metodom startModule().



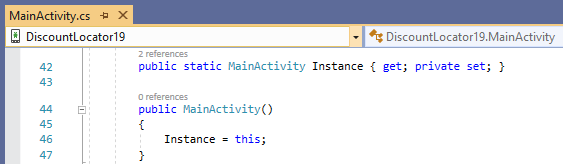
Još se moramo vratiti u ListViewModule klasu. U displayData() metodu promijenit ćemo kontekst koji se šalje adapteru za expandable recycler view.



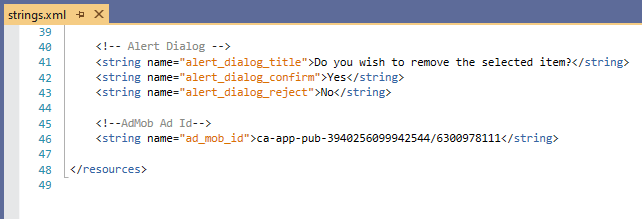
Također se moramo vratiti u TitleChildViewHolder kako bismo izmijenili metodu koja se poziva na duži klik. Pozvat ćemo startMainModule() metodu koja će ponovno prikazati podatke u listi, ovaj put ažurirane jer smo obrisali odabrani popust. Također smo promijenili kontekst kod kreiranja AlertDialog objekta. Prosljeđujemo instancu MainActivity klase. Dodat ćemo i prijevod poruke za brisanje popusta.



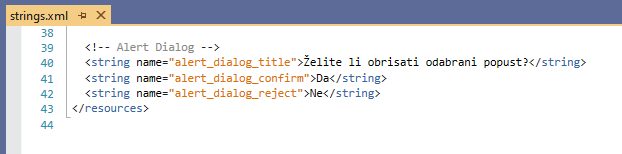
U MainActivity klasu moramo dodati Instance() metodu koju smo koristili prilikom kreiranja AlertDialog-a.



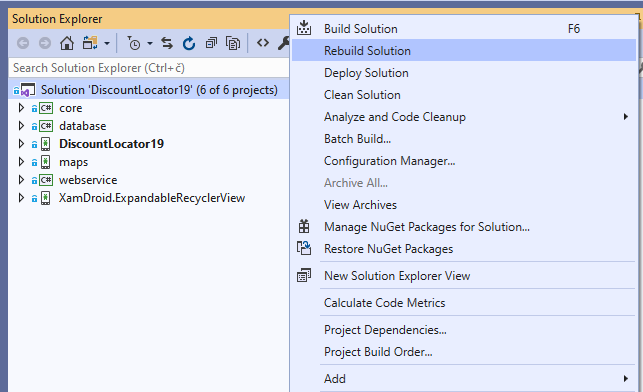
Prijevode ćemo dodati u strings.xml datotekama. DiscountLocator19 -> Resources -> values -> strings.xml:



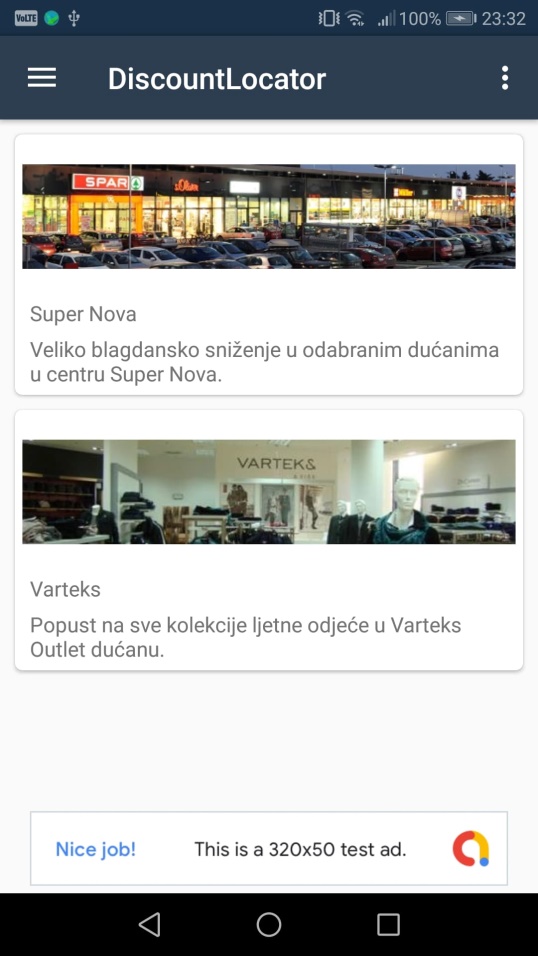
DiscountLocator19 -> Resources -> values-hr -> strings.xml:



Prije testiranja moramo napraviti Rebuild cijelog projekta.



Na pokretanje aplikacije podaci će nam se prikazati u listi (expandable recycler view).



Glavni izbornik otvaramo klikom na ikonu u gornjem lijevom kutu ekrana. Odabiremo prikaz na mapi. 