

```
# Sistem preporuka u aplikaciji NovinskiPortal
```

U aplikaciji NovinskiPortal koristi se jednostavan content based sistem preporuke za članke. Sistem preporuke radi samo za prijavljene korisnike. Za svakog korisnika prati koje kategorije članaka najčešće čita i na osnovu toga predlaže nove, još nepročitane članke iz tih kategorija.

1. Podaci koje sistem koristi

Sistem preporuke koristi sljedeće podatke iz baze:

- *UserCategoryPreference*
Za svakog korisnika pamti koju kategoriju čita i koliko često. Bitna polja su UserId, CategoryId, ViewCount i LastViewedAt.
- *UserArticleView*
Za svakog korisnika pamti koje je članke otvorio. Bitna polja su UserId, ArticleId i ViewedAt.
- *Article* i *ArticleStatistics*
Članak ima CategoryId, SubcategoryId, PublishedAt i Active.
U *ArticleStatistics* se čuva ukupan broj pregleda članka (TotalViews).

2. Kako sistem preporuke radi

Glavna metoda je *RecommendationService.GetPersonalizedAsync(userId, take)*. Logika rada ukratko:

1. Pronalaženje omiljenih kategorija korisnika
 - Iz tabele *UserCategoryPreferences* uzimaju se sve preferencije za datog korisnika.
 - Sortiraju se po ViewCount opadajuće.
 - Uzimaju se najviše 3 kategorije koje korisnik najviše čita.

Ako korisnik nema nijednu preferencu, trenutno se vraća prazna lista preporuka.

2. Isključenje već pročitanih članaka

- Iz tabele *UserArticleView* uzimaju se svi ArticleId koje je korisnik već otvorio.

- Ti članci se kasnije izbacuju iz rezultata da se ne preporučuju ponovo.

3. Odabir i rangiranje preporuka

- Iz tabele *Articles* uzimaju se samo aktivni članci iz korisnikovih top kategorija.
- Ako postoji lista već pročitanih članaka, oni se isključuju iz upita.
- Članci se sortiraju po PublishedAt opadajuće, pa po TotalViews opadajuće.
- Uzimaju se najviše take članaka i mapiraju u *ArticleResponse*.

4. Fallback u slučaju da nema rezultata

- Ako nakon filtriranja nema nijednog članka, ponovo se uzimaju članci iz istih kategorija.
- Sortiraju se prvo po TotalViews opadajuće, zatim po PublishedAt.
- Tako korisnik dobije najčitanije članke iz svojih omiljenih kategorija.

3. Implementacija u kodu

Servis:

- Namespace: *NovinskiPortal.Services.Services.RecommendationService*
- Klasa: *RecommendationService*
- Metoda: *Task<List<ArticleResponse>> GetPersonalizedAsync(int userId, int take = 10)*

Entiteti:

- *UserCategoryPreference* i *UserArticleView* u namespaceu *NovinskiPortal.Services.Database.Entities*.
Ovi entiteti se koriste za bilježenje preferencija po kategorijama i historije pregleda članaka.

API:

- Namespace: *NovinskiPortal.API.Controllers*
- Klasa: *RecommendationController*
- Endpoint:
GET/api/recommendation/personalized?take=10

Kontroler čita userId iz JWT tokena, poziva *IRecommendationService.GetPersonalizedAsync* za tog korisnika i vraća listu preporučenih članaka kao *List<ArticleResponse>*.

```
public async Task<List<ArticleResponse>> GetPersonalizedAsync(int userId, int take = 10)
{
    var topCategories = await _context.UserCategoryPreferences
        .Where(x => x.UserId == userId)
        .OrderByDescending(x => x.ViewCount)
        .Select(x => x.CategoryId)
        .Take(3)
        .ToListAsync();

    if (!topCategories.Any())
    {
        return new List<ArticleResponse>();
    }

    var viewedArticleIds = await _context.UserArticleViews
        .Where(v => v.UserId == userId)
        .Select(v => v.ArticleId)
        .Distinct()
        .ToListAsync();

    var query = _context.Articles
        .Include(a => a.Category)
        .Include(a => a.Subcategory)
        .Include(a => a.User)
        .Include(a => a.Statistics)
        .Where(a => a.Active)
        .Where(a => topCategories.Contains(a.CategoryId));

    if (viewedArticleIds.Any())
    {
        query = query.Where(a => !viewedArticleIds.Contains(a.Id));
    }

    var articles = await query
        .OrderByDescending(a => a.PublishedAt)
        .ThenByDescending(a => a.Statistics != null ? a.Statistics.TotalViews : 0)
        .Take(take)
        .ToListAsync();

    if (articles.Any())
    {
        return _mapper.Map<List<ArticleResponse>>(articles);
    }

    var mostReadInTopCategories = await _context.Articles
        .Include(a => a.Category)
        .Include(a => a.Subcategory)
        .Include(a => a.User)
        .Include(a => a.Statistics)
        .Where(a => a.Active)
        .Where(a => topCategories.Contains(a.CategoryId))
        .OrderByDescending(a => a.Statistics != null ? a.Statistics.TotalViews : 0)
        .ThenByDescending(a => a.PublishedAt)
        .Take(take)
        .ToListAsync();

    return _mapper.Map<List<ArticleResponse>>(mostReadInTopCategories);
}
```



Detaljnije o ovom istraživanju možete pročitati ovdje.

Možda vas zanima



Inspekcije "pročešljale" socijalne ustanove u Sar...



Istanbul je najpoželjniji grad Evrope za ovu godin...



Objavljena lista najboljih otoka u Evropi u 2025. g...



Snijeg je nakratko zabijelio Sarajevo: Pogle...



Pogledajte kako izgleda Vjesnikov neboder u Zag...



Velika akcija u Srbiji, uhapšeno 18 osoba: Kriv...