Recommender sistemi u NutriTrack

Ovaj dokument opisuje **obicni** (`GroceryRecommender`) i **napredni** (`EnhancedGroceryRecommender`) recommender. Oba sistema preporucuju namirnice koristeci podatke o korisniku, njegov cilj ishrane, popularnost namirnica i sezonalnost.

1) Obicni recommender (`GroceryRecommender`)

Tok rada

1. Ucita korisnika i njegov aktivni cilj ishrane.
2. Dohvati do **1000** kandidata iz baze (namirnice).
3. Izracuna popularnost namirnica unutar korisnickog kohorta (korisnici s istim ciljem).
4. Generise feature set i skorira ih ML/heuristikom.
5. Primjenjuje novelty i cold-start boost.
6. Sortira i paginira rezultate.

Ključni dijelovi koda

Kohort i popularnost:

```
var cohortUserIds = await GetCohortUserIdsAsync(goalType, cancellationToken);
var popularityByGroceryId = await _mealItemReadRepository.CountByGroceryForUsersAsync(
    cohortUserIds.Select(id => new UserId(id)).ToArray(),
    cancellationToken);
```

Feature set + ML scoring:

```
var features = GroceryFeatures.FromGrocery(
    g,
    popularityCount,
    goalType,
    userAge,
    (int)user.Gender,
    (int)user.ActivityLevel,
    seasonalityScore);

var score = _mlScorer.Score(features);
```

Boosting logika:

```
var noveltyBoost = popularityCount < 5 ? 1.3 : 1.0;
var coldStartBoost = GetColdStartBoost(user, popularityCount);

return new
{
    Grocery = g,
```

```
    Score = score * noveltyBoost * coldStartBoost
};
```

2) Napredni recommender (`EnhancedGroceryRecommender`)

Sta dodaje u odnosu na obicni

- **Diversity manager**: penalizira previse istih kategorija.
- **Eksplanacije**: objasnjava zasto je namirnica preporucena.
- **Praćenje metrika**: biljezi koji preporuceni item je prikazan.
- **Korisnicka istorija (30 dana)**: radi boljih objasnjenja.

Ključni dijelovi koda

Diversity + scoring:

```
var noveltyBoost = popularityCount < NoveltyBoostThreshold ? NoveltyBoostMultiplier : 1.0;
var diversityPenalty = _diversityManager.GetDiversityPenalty(grocery.Id.Value, grocery.Categ
var coldStartBoost = GetColdStartBoost(user, popularityCount);

var finalScore = baseScore * noveltyBoost * diversityPenalty * coldStartBoost;
```

Generisanje objasnjenja:

```
var explanation = BuildExplanation(
    grocery,
    goalType,
    popularityCount,
    seasonalityScore,
    userRecentGroceries.Contains(grocery.Id.Value));
```

Tracking prikazanih preporuka:

```
await _metricsTracker.TrackRecommendationShownAsync(
    userId.Value,
    pageItems[i].Id.Value,
    pageItems[i].Score,
    position,
    cancellationToken);
```

3) ML scoring (ML.NET + fallback heuristika)

Ako model nije treniran, koristi se fallback heuristika:

```
var macroScore = goalType switch
{
    NutritionGoal.LoseWeight =>
        (features.ProteinPer100 * 2.0) -
        (features.CaloriesPer100 * 0.02) -
        (features.FatPer100 * 0.5) -
```

```
        (features.CarbsPer100 * 0.2),

    NutritionGoal.GainWeight =>
        (features.CaloriesPer100 * 0.02) +
        (features.CarbsPer100 * 0.6) +
        (features.FatPer100 * 0.8) +
        (features.ProteinPer100 * 0.8),

    _ =>
        (features.ProteinPer100 * 1.0) +
        (features.CarbsPer100 * 0.3) +
        (features.FatPer100 * 0.3) -
        (Math.Abs(features.CaloriesPer100 - 200) * 0.01)
};
```

4) Sažetak

- **Obicni recommender**: fokus na popularnost + ML skor + novelty/cold-start boost.
- **Napredni recommender**: dodaje diversitet, objasnjenja i tracking metrika.
- **ML sloj**: koristi ML.NET (SDCA regresija) ili fallback heuristiku.