Amar Omerović

IB210242
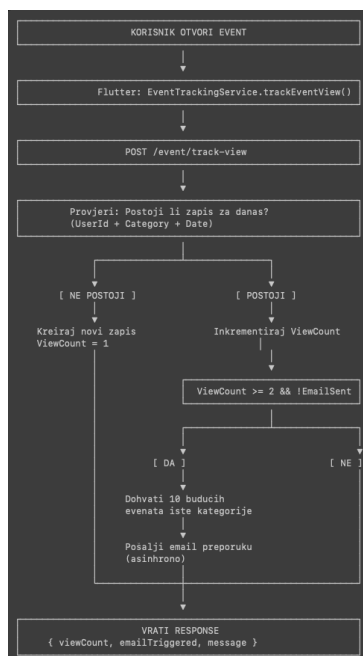
Karta.ba – Recommender Dokumentacija

1. Opis sistema preporuke

Core Logic

Sistem analizira ponašanje korisnika prilikom pregleda događaja i koristi content-based pristup za preporuke. Umjesto složenih machine learning algoritama, sistem koristi jednostavnu rule-based logiku baziranu na kategorijama događaja.

Ključni proračuni

- Za svakog korisnika prati se broj pregleda događaja po kategoriji u jednom danu

- Kada korisnik pregleda 2 ili više događaja iste kategorije, sistem automatski šalje email sa preporukama

- Email sadrži top 10 budućih događaja iz iste kategorije

- Prilikom testiranja je potrebno 2 puta otvoriti vec postojeci event

Primjena

Korisnik automatski prima personalizovane email preporuke. Organizator dobija bolju vidljivost svojih događaja.
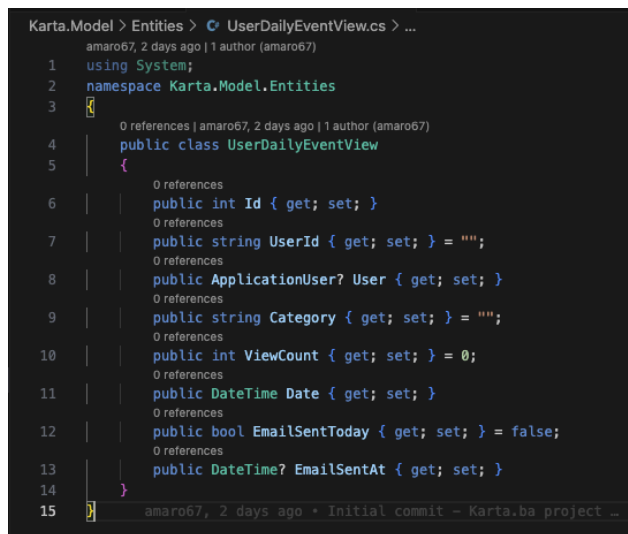
2. Opis implementacije

Tehnologije

Backend: ASP.NET Core, EF Core, SQL Server

UI: Flutter

Email: SMTP / RabbitMQ

```
Karta.Model > Entities > C# UserDailyEventView.cs > …
        amaro67, 2 days ago | 1 author (amaro67)
  1   using System;
  2   namespace Karta.Model.Entities
  3   {
            0 references | amaro67, 2 days ago | 1 author (amaro67)
  4       public class UserDailyEventView
  5       {
                0 references
  6           public int Id { get; set; }
                0 references
  7           public string UserId { get; set; } = "";
                0 references
  8           public ApplicationUser? User { get; set; }
                0 references
  9           public string Category { get; set; } = "";
                0 references
 10           public int ViewCount { get; set; } = 0;
                0 references
 11           public DateTime Date { get; set; }
                0 references
 12           public bool EmailSentToday { get; set; } = false;
                0 references
 13           public DateTime? EmailSentAt { get; set; }
 14       }
 15   }         amaro67, 2 days ago • Initial commit - Karta.ba project …
```

3. Ključni API endpointi i algoritam

Algoritam se aktivira kada korisnik pregleda 2+ događaja iste kategorije u jednom danu.

4. Sigurnost

JWT autentikacija i [Authorize] atribut.

```
[HttpGet("my-events")]
[Authorize]
[SwaggerOperation(Summary = "Dohvata sve evente koje je kreirao trenutni korisnik",
                  Description = "Vraća listu eventa koje je kreirao organizator, sortirane po datumu kreiranja (najnoviji prvi)")]
[SwaggerResponse(200, "Uspešno vraćena lista eventa", typeof(IReadOnlyList<EventDto>))]
[SwaggerResponse(401, "Korisnik nije autentifikovan")]
0 references
public async Task<ActionResult<IReadOnlyList<EventDto>>> GetMyEvents()
```

5. Komunikacija

Flutter aplikacija komunicira sa backendom putem HTTP zahtjeva uz JWT token.

```dart
karta_UI > karta_mobile > lib > services > event_tracking_service.dart
You, now | 2 authors (amaro67 and one other)
1    import 'dart:convert';
2    import 'package:karta_shared/karta_shared.dart';
3    class EventTrackingService {
4      static Future<Map<String, dynamic>?> trackEventView(String eventId, {String? token}) async {
5        try {
6          print('📊 Tracking event view: $eventId');
7          final data = await ApiClient.post(
8            '/event/track-view',
9            {'eventId': eventId},
10           token: token,
11         );
12         print('Event view tracked: $data');
13         if (data['emailTriggered'] == true) {
14           print('Email triggered! User will receive recommendations.');
15         }
16         return data;
17       } catch (e) {
18         print('Error tracking event view: $e');
19         return null;
20       }
21     }
22   }         amaro67, 2 days ago • Initial commit — Karta.ba project …
```
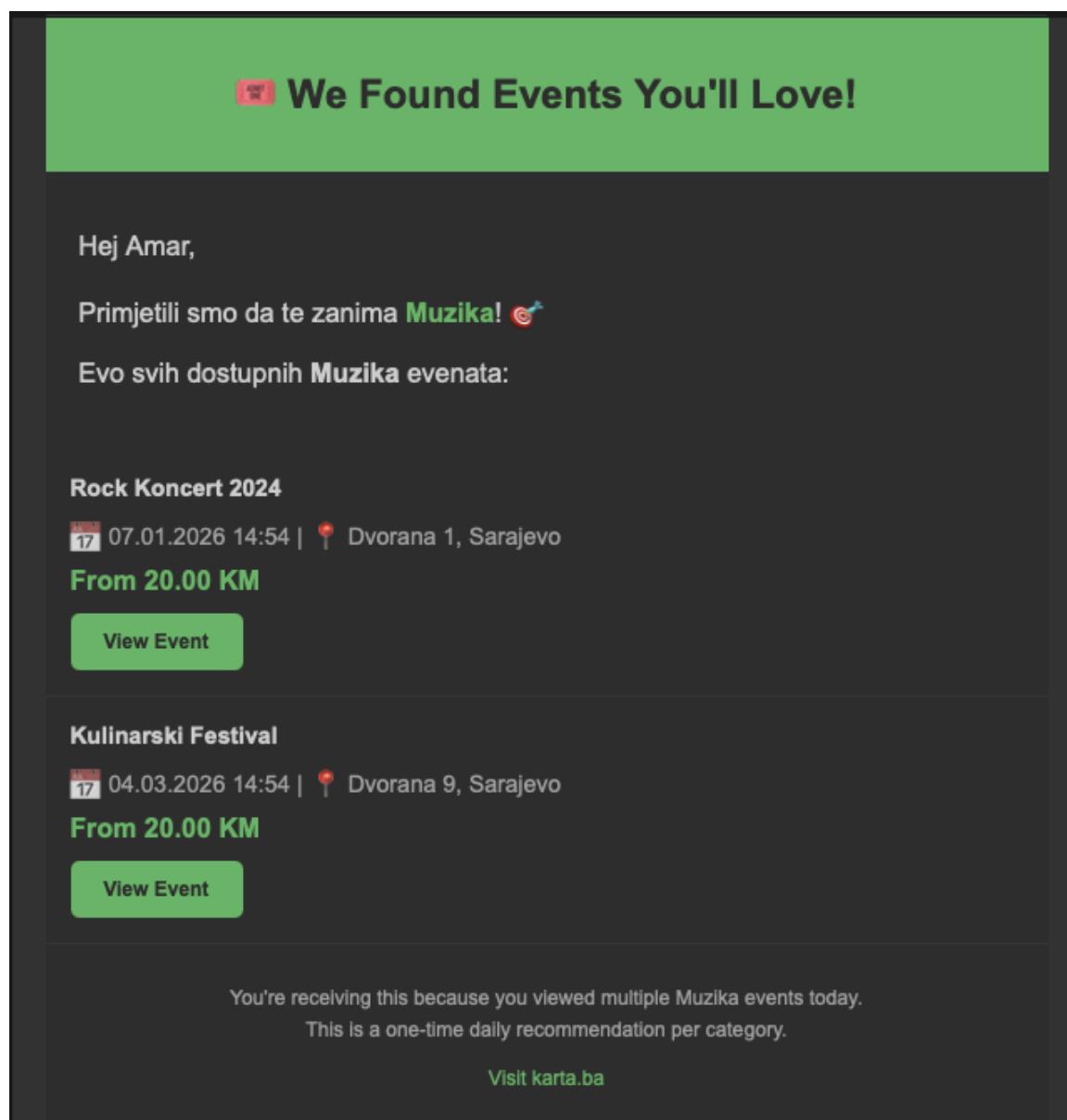
6. Email preporuke

```csharp
var categoryEvents = await _context.Events
    .Include(e => e.PriceTiers)
    .Where(e =>
        e.Category == eventItem.Category &&
        e.Status == "Published" &&
        e.StartsAt > DateTimeOffset.UtcNow)
    .OrderBy(e => e.StartsAt)
    .Take(10)
    .ToListAsync();
if (categoryEvents.Any())
{
    dailyView.EmailSentToday = true;
    dailyView.EmailSentAt = DateTime.UtcNow;
    await _context.SaveChangesAsync();
    var userEmail = user.Email;
    var userName = user.FirstName ?? userEmail.Split('@')[0];
    var category = eventItem.Category;
    _ = Task.Run(async () =>
    {
        try
        {
            var emailBody = GenerateCategoryRecommendationEmailBody(userName, category, categoryEvents);
            await _emailService.SendEmailDirectAsync(
                userEmail,
                $"{category} Events You'll Love!",
                emailBody
            );         amaro67, 2 days ago • Initial commit — Karta.ba project
            _logger.LogInformation($"Email sent directly to {userEmail} - Category: {category}, Events: {categoryEvents.Count}");
        }
        catch (Exception ex)
        {
            _logger.LogError(ex, $"Error sending category recommendation email for user {userId}");
        }
    });
}
```

8. UI orkestracija (mobile)

```
@override
void didChangeDependencies() {          amaro67, 2 days ago • Initial commit — Karta.ba project …
  super.didChangeDependencies();
  if (!_hasRecordedView) {
    final event = ModalRoute.of(context)!.settings.arguments as EventDto;
    print('EventDetailScreen: Recording view for event: ${event.id} — ${event.title}');
    ViewedEventsService.addViewedEvent(event.id);
    final authProvider = Provider.of<AuthProvider>(context, listen: false);
    if (authProvider.isAuthenticated && authProvider.accessToken != null) {
      EventTrackingService.trackEventView(
        event.id,
        token: authProvider.accessToken,
      );
    } else {
      print('User not authenticated — skipping backend tracking');
    }
    _hasRecordedView = true;
    print('EventDetailScreen: View recorded successfully');
  }
}
```

9. Vizuelne preporuke

10. Struktura fajlova

Komponenta	Putanja
Database Model	Karta.Model/Entities/UserDailyEventView.cs
DB Context Config	Karta.Model/ApplicationDbContext.cs
API Endpoint	Karta.WebAPI/Controllers/EventController.cs
Email Service	Karta.Service/Services/EmailService.cs
Email DTO	Karta.Service/DTO/EmailMessage.cs

Komponenta	Putanja
Daily Reset Service	Karta.WebAPI/Services/DailyResetService.cs
Flutter Tracking	karta_mobile/lib/services/event_tracking_service.dart
Flutter Screen	karta_mobile/lib/screens/user/event_detail_screen.dart
Migration	Karta.WebAPI/Migrations/20251208175317_AddUserDailyEventView.cs

11. Ključne metrike

Metrika	Vrijednost
Vremenski prozor	24 sata (UTC dan)
Trigger prag	2+ pregleda iste kategorije
Max preporuka po emailu	10 događaja
Email frekvencija	Max 1 po kategoriji po danu
Čišćenje starih zapisa	Automatski u ponoć UTC

12. Primjer scenarija

Scenario 1: Korisnik zainteresovan za muziku

1. **10:00** - Korisnik otvori koncert "Rock Festival" (kategorija: Music)
   - ViewCount = 1, emailTriggered = false
2. **14:00** - Korisnik otvori "Jazz Night" (kategorija: Music)
   - ViewCount = 2, emailTriggered = true
   - Sistem šalje email sa 10 budućih Music događaja
3. **18:00** - Korisnik otvori "Classical Concert" (kategorija: Music)
   - ViewCount = 3, emailTriggered = false
   - Poruka: "Email already sent today"

Scenario 2: Više kategorija

1. Korisnik pregleda 2 Music događaja → dobije Music preporuke
2. Isti dan pregleda 2 Sports događaja → dobije Sports preporuke
3. Svaka kategorija ima nezavisan brojač i email limit

Zaključak

Sistem predstavlja jednostavan i efikasan content-based recommender koji automatski šalje personalizovane email preporuke.