

# eBarber - Funkcionalna Specifikacija

## 1. UVOD

### 1.1 Opis aplikacije

eBarber je mobilna aplikacija za digitalizacijo poslovanja frizerskih salonov. Omogoča: - Strankami: enostavno rezervacijo terminov, pregled zgodovine, ocenjevanje - Frizerjem (Admin): upravljanje terminov, strank, ocen in komunikacije

### 1.2 Glavni cilji

- Poenostavitev naročanja (brez telefonskih klicev)
  - Avtomatsko razporejanje terminov glede na kakovost strank
  - Centralizirana komunikacija med frizerjem in strankami
  - Upravljanje dveh lokacij (Dom na Poljanski cesti, Frankolovo)
- 

## 2. ARHITEKTURA SISTEMA

### 2.1 Tehnični sklad

**Backend:** - Framework: ASP.NET Core 8 (C#) - Baza podatkov: SQL Server (Docker) - ORM: Entity Framework Core - Avtentikacija: JWT Bearer tokens + ASP.NET Identity

**Mobile App:** - Framework: React Native (Expo) - Navigacija: React Navigation - API komunikacija: Axios - Secure Storage: expo-secure-store

### 2.2 Komponente sistema

Mobile App  
(React Native)

HTTPS/JWT

.NET Core API  
(REST API)

Entity Framework

SQL Server DB  
(Docker)

---

### 3. PODATKOVNI MODEL

#### 3.1 ER Diagram

ApplicationUser (AspNetUsers)

- Id (PK)
- Email
- FirstName, LastName
- TotalBookings
- CompletedBookings
- CancelledBookings
- AverageRating (decimal)
- ReliabilityScore (computed)

Lokacija

- Id (PK)
- Naslov
- Mesto
- DelovniCas

OkvirniTermin

- Id (PK)
- ZacetekCasa (datetime)
- KonecCasa (datetime)
- LokacijaId (FK)
- MaksimalnoUporabnikov (int)
- Opis (string)
- JeAktiven (bool)

PrijavaTermina

- Id (PK)
- OkvirniTerminId (FK)
- UprabnikId (FK)
- DatumPrijava (datetime)
- JePrispel (bool)
- OcenaStranke (int 1-5)
- Status (string: Potrjena/Preklicana)
- Opombe (string)

Message

- Id (PK)
- SenderId (FK)
- ReceiverId (FK, nullable)
- Content (string)
- SentAt (datetime)
- IsRead (bool)
- MessageType (string)

### 3.2 Relacije

- User → PrijavaTermina: One-to-Many
  - OkvirniTermin → PrijavaTermina: One-to-Many
  - Lokacija → OkvirniTermin: One-to-Many
  - User → Message: One-to-Many (sender/receiver)
- 

## 4. PAMETNI ALGORITEM RAZPOREJANJA

### 4.1 Opis problema

Ko je okvirni termin poln (npr. 5/5 mest), in se želi prijaviti nova stranka: - **Tradicionalno (FIFO)**: Nova stranka ne more - "Poln termin" - **eBarber Smart Scheduling**: Primerjamo priority score in zamenjamo najslabšo stranko

### 4.2 Priority Score Formula

Priority Score = (Rating Score × 0.4) + (Reliability Score × 0.4) + (Visit Score × 0.2)

Kjer:

- Rating Score = AverageRating / 5 // 0.0 - 1.0
- Reliability Score = CompletedBookings / TotalBookings // 0.0 - 1.0
- Visit Score = min(TotalBookings / 10, 1.0) // max 1.0 pri 10+ obiskih

### 4.3 Primer izračuna

**Stranka A:** - AverageRating: 4.5/5 → Rating Score = 0.9 - CompletedBookings: 8, TotalBookings: 10 → Reliability = 0.8 - TotalBookings: 10 → Visit Score = 1.0

Priority Score A = (0.9 × 0.4) + (0.8 × 0.4) + (1.0 × 0.2)  
= 0.36 + 0.32 + 0.2 = 0.88

**Stranka B:** - AverageRating: 2.0/5 → Rating Score = 0.4 - CompletedBookings: 3, TotalBookings: 6 → Reliability = 0.5 - TotalBookings: 6 → Visit Score = 0.6

Priority Score B = (0.4 × 0.4) + (0.5 × 0.4) + (0.6 × 0.2)  
= 0.16 + 0.2 + 0.12 = 0.48

**Rezultat:** Stranka A ima višji score (0.88 > 0.48), zato dobi prednost.

### 4.4 Algoritem (C# psevdokoda)

```
if (termin.JePoln())
{
    novPriorityScore = IzracunajScore(novaStranka);
    najslabsaStranka = NajdiNajslabso(termin.Prijave);
    najslabsiScore = IzracunajScore(najslabsaStranka);

    if (novPriorityScore > najslabsiScore)
    {
        PrekliciPrijava(najslabsaStranka);
    }
}
```

```

        DodajPrijava(novaStranka);
        return "Uspeh - zamenjava";
    }
    else
    {
        return "Zavrnen - prenizek score";
    }
}
else
{
    DodajPrijava(novaStranka);
    return "Uspeh";
}

```

---

## 5. API DOKUMENTACIJA

### 5.1 Avtentikacija

#### POST /api/auth/login

##### Request:

```

{
    "email": "stranka@test.si",
    "password": "Test123!"
}

```

##### Response (200 OK):

```

{
    "token": "eyJhbGciOiJIUzI1NiIs... ",
    "email": "stranka@test.si",
    "firstName": "Test",
    "lastName": "Stranka"
}

```

#### POST /api/auth/register

##### Request:

```

{
    "email": "nova@stranka.si",
    "password": "Geslo123!",
    "firstName": "Nova",
    "lastName": "Stranka"
}

```

##### Response (200 OK):

```

{
    "message": "Registracija uspešna"
}

```

## 5.2 Okvirni termini

GET /api/OkvirniTerminiApi

Response (200 OK):

```
[
  {
    "id": 7,
    "zacetekCasa": "2026-01-20T17:00:00",
    "konecCasa": "2026-01-20T20:00:00",
    "opis": "Testni termin",
    "maksimalnoUporabnikov": 5,
    "lokacija": {
      "naslov": "Dom na Poljanski cesti",
      "mesto": "Ljubljana"
    },
    "steviloPrijav": 2,
    "jePolno": false
  }
]
```

POST /api/OkvirniTerminiApi/{id}/prijava

Request:

```
{
  "opombe": "Želim kratko striženje"
}
```

Response (200 OK - normalna prijava):

```
{
  "message": "Uspešno ste se prijavili na termin"
}
```

Response (200 OK - smart scheduling):

```
{
  "message": "Prijava uspešna! Zaradi višjega priority score-a ste dobili mesto.",
  "priorityScore": 0.88,
  "replacedUser": "slaba@stranka.si"
}
```

Response (400 Bad Request):

```
{
  "message": "Termin je poln. Vaš priority score ni dovolj visok.",
  "yourScore": 0.45,
  "requiredScore": 0.60
}
```

## 5.3 Admin endpoints

**\*\*GET /api/OkvirniTerminiApi/admin/all\*** *Zahteva: Role = Administrator*

**Response:**

```
[
  {
    "id": 7,
    "zacetekCasa": "2026-01-20T17:00:00",
    "prijave": [
      {
        "id": 1,
        "jePrispel": false,
        "ocenaStranke": null,
        "uporabnik": {
          "firstName": "Test",
          "lastName": "Stranka",
          "email": "stranka@test.si",
          "averageRating": 4.5
        }
      }
    ]
  }
]
```

**POST /api/OkvirniTerminiApi/admin/checkin/{prijavaId}**

**Response (200 OK):**

```
{
  "message": "Check-in uspešen"
}
```

**POST /api/OkvirniTerminiApi/admin/rate/{prijavaId}**

**Request:**

```
{
  "rating": 5
}
```

**Response (200 OK):**

```
{
  "message": "Ocena shranjena"
}
```

**GET /api/admin/users**

**Response:**

```
[
  {
    "id": "15ec8cda-...",
    "email": "stranka@test.si",
    "firstName": "Test",
    "lastName": "Stranka",
    "totalBookings": 10,
    "completedBookings": 9,
  }
]
```

```
    "cancelledBookings": 1,  
    "averageRating": 4.5  
  }  
]
```

POST /api/messages

Request:

```
{  
  "content": "Lahko prestavim termin?"  
}
```

Response (200 OK):

```
{  
  "message": "Sporočilo poslano"  
}
```

---

## 6. UPORABNIŠKI TOK (USER FLOW)

### 6.1 Stranka - Rezervacija termina

```
START  
↓  
[Login Screen] → Vnos email/geslo  
↓  
[Moji Termini] → Pregled preteklih rezervacij  
↓  
Klik na "Okvirni termini"  
↓  
[Okvirni Termini] → Seznam razpoložljivih terminov  
↓  
Izbira termina → Klik "Prijava se"  
↓  
Algoritem preveri score → Potrditev/Zavrnitev  
↓  
[SUCCESS] → "Uspešno rezervirano!"  
↓  
END
```

### 6.2 Admin - Check-in in ocenjevanje

```
START  
↓  
[Login kot admin]  
↓  
Klik "Admin Panel"  
↓  
Izbira: [Termini] ali [Stranke] ali [Dodaj termin]
```

↓  
[Admin Termini] → Seznam vseh terminov + prijave  
↓  
Klik "Check-in" → Uporabnik označen kot prispel  
↓  
Klik "Oceni" → Vnos ocene 1-5  
↓  
Sistem izračuna novo povprečno oceno  
↓  
Posodobi reliability score  
↓  
END

---

## 7. ZASLONSKE MASKE

### 7.1 Mobile App - Stranka

**Login Screen:** - Email input field - Password input field (secure) - "Prijava" button - Link na registracijo

**Moji Termini Screen:** - Header z uporabniškim emailom - Seznam preteklih terminov (datum, lokacija, status) - FAB gumbi: - Okvirni termini - Sporočila - Nov termin

**Okvirni Termini Screen:** - Seznam razpoložljivih terminov - Prikaz: datum, čas, lokacija, zasedenost (2/5) - "Prijavi se" gumb (disabled če poln) - Pull-to-refresh

**Nova Rezervacija Screen:** (*opcijsko - direktna izbira*) - Izbira lokacije - Izbira datuma (YYYY-MM-DD) - Izbira ure (HH:MM) - Opombe textarea - "Rezerviraj" button

### 7.2 Mobile App - Admin

**Admin Menu:** - Okvirni termini - Vse stranke - Dodaj okvirni termin

**Admin Termini Screen:** - Seznam vseh okvirnih terminov - Za vsak termin: - Datum, čas, lokacija - Seznam prijavljenih uporabnikov - Check-in gumb (za vsakega uporabnika) - Oceni gumb (1-5 zvezdic)

**Admin Stranke Screen:** - Seznam vseh registriranih uporabnikov - Prikaz: - Ime, email - Povprečna ocena (4.5) - Zanesljivost (80%) - Klik → Podrobnosti stranke

**Admin User Detail Screen:** - Uporabniški podatki (ime, email) - Statistika: - Skupaj terminov - Prihodi / Preklici - Zanesljivost % - Obrazec za ročno nastavitev ocene (0.0 - 5.0) - Zgodovina terminov (seznam)

**Admin Create Termin Screen:** - Izbira lokacije - Datum (YYYY-MM-DD) - Začetek (HH:MM) - Konec (HH:MM) - Max uporabnikov (number input) - Opis (textarea) - "Ustvari termin" button

### 7.3 Web Admin Panel

**Dashboard:** - Kartice s statistiko: - Število uporabnikov - Današnji termini - Nova sporočila - Povprečna ocena - Gumbi: - Upravljanje strank - Okvirni termini - Sporočila



**Users Screen:** - Tabela: - Ime, Email - Termini (skupaj) - Prihodi, Preklici - Zanesljivost % - Povprečna ocena - "Podrobnosti" button

**User Details Screen:** - Kartica z uporabnikom - Obrazec za nastavitev ocene - Obrazec za posodobitev statistike - Tabela preteklih terminov: - Datum, Lokacija - Prispel? (ja/ne) - Ocena (če obstaja) - Check-in button (če ni prispel) - Ocen dropdown (1-5 zvezdic)

---

## 8. TESTIRANJE

### 8.1 Unit testi

- **Priority Score Calculation:** Preveri pravilnost formule
- **User Authentication:** JWT token generation/validation
- **Database Queries:** CRUD operacije

### 8.2 Integracijski testi

- **Smart Scheduling:** Scenarij polnega termina
- **API Endpoints:** Request/Response validacija
- **Mobile Backend:** End-to-end flow

### 8.3 Uporabniški scenariji

**Scenarij 1: Nova stranka z dobrim scorom** 1. Nova stranka se registrira 2. Admin ji ročno nastavi oceno 4.8 3. Stranka se prijavi na poln termin (5/5) 4. Algoritem jo primerja z najslabšo stranko (score 0.45) 5. Stranka dobi mesto

**Scenarij 2: Nova stranka z nizkim scorom** 1. Nova stranka brez ocene (default 0.0) 2. Poskuša se prijaviti na poln termin 3. Algoritem jo primerja (score 0.0 vs najslabši 0.60) 4. Stranka je zavrnjena

**Scenarij 3: Admin Check-in in ocenjevanje** 1. Admin odpre "Admin Panel" → "Termini" 2. Vidi seznam prijavljenih (Test Stranka) 3. Klikne → Check-in uspešen 4. Klikne zvezdico → Vnese oceno 5 5. Sistem posodobi povprečno oceno uporabnika

---

## 9. VARNOST

### 9.1 Avtentikacija

- JWT Bearer tokens (30-day expiry)
- Password hashing (ASP.NET Identity default)
- Secure storage v mobile app (expo-secure-store)

### 9.2 Avtorizacija

- Role-based access control (Administrator / User)
- API endpoints zaščiteni z [Authorize] atributom
- Admin-only endpoints: [Authorize(Roles = "Administrator")]

## 9.3 Validacija

- Backend: Data annotations (Required, StringLength, Range)
  - Frontend: Input validation pred submit
  - SQL injection protection: Entity Framework parameterized queries
- 

## 10. DEPLOYMENT

### 10.1 Backend

#### Development:

```
cd ~/eBarber3/eBarber
dotnet run
```

Teče na: <http://localhost:5126>

**Production:** - Azure App Service - SQL Azure Database - Environment variables za Jwt:Key

### 10.2 Mobile App

#### Development:

```
cd ~/eBarberMobile
npx expo start
```

**Production:** - Expo EAS Build - Export APK za Android - TestFlight/App Store za iOS

### 10.3 Database

#### Docker (development):

```
docker run -e "ACCEPT_EULA=Y" -e "SA_PASSWORD=YourStrong@Passw0rd" \
  -p 1433:1433 --name sqlserver -d mcr.microsoft.com/mssql/server:2022-latest
```

#### Connection String:

Server=localhost,1433;Database=eBarberDB;User=sa;Password=YourStrong@Passw0rd;TrustServerCertificate=true

---

## 11. ZAKLJUČEK

Aplikacija eBarber uspešno implementira: - Digitalizacijo rezervacij terminov - Pametno razporejanje glede na kakovost strank - Admin panel za upravljanje - Sistem ocenjevanja in sledenja metrikam - Osnovno komunikacijo (sporočila) - Mobilno in spletno platformo

#### Prihodnje izboljšave:

- Push notifications za potrjene/preklicane termine
- Integracija s koledarjem
- Plačilni sistem
- Analitika z grafi (Charts.js)

- Multi-jezik podpora

---

**Konec funkcionalne specifikacije**