

# Developer Guide — Ishnka

Restaurant Platform Engineering

## Project Responsibility Summary

You are responsible for authentication, guest flow, profiles, CRM events, and push notifications.

- Secure JWT auth
- Guest login tied to QR/table
- User profile preferences (language, dietary)
- Favorites and visits
- Notification subscription and sending
- MongoDB logs for notification events

## Technology Stack

- Spring Boot
- MySQL (users, tokens, favorites, visits)
- MongoDB (notification events)
- React (Auth + Profile + Notifications UI)

## Database Responsibilities

### MySQL Tables

```
users(id, email, password_hash, name)
refresh_tokens(id, user_id, token_hash, revoked)
user_profile(user_id, language, dietary json, preferences json)
favorites(user_id, item_id)
visits(id, user_id, restaurant_id, session_id, visited_at)
subscriptions(id, user_id, endpoint, p256dh, auth)
notification_log(id, user_id, type, payload json, status, created_at)
```

### MongoDB Collections

```
notification_events {
  userId,
  event,
  status,
  timestamp,
  payload{}}
```

# API Endpoints

## Auth

```
POST /api/v1/auth/signup  
POST /api/v1/auth/login  
POST /api/v1/auth/guest  
POST /api/v1/auth/refresh  
POST /api/v1/auth/logout  
GET  /api/v1/auth/verify
```

## Profile + Favorites

```
GET   /api/v1/users/{userId}  
PATCH /api/v1/users/{userId}  
POST  /api/v1/users/{userId}/favorites/{itemId}  
DELETE /api/v1/users/{userId}/favorites/{itemId}
```

## Notifications

```
POST /api/v1/notifications/web  
POST /api/v1/notifications/email  
POST /api/v1/notifications/sms
```

# Backend Development Guidelines

## Auth Rules

- BCrypt password hashing
- JWT access token (15 min)
- Refresh token rotation stored in MySQL
- Guest login returns temp token linked to table/session

## Security Requirements

- Validate input at DTO level
- Prevent brute force (Redis throttle)
- Never store plain tokens
- Secure same-site cookie option possible

## MongoDB Logging Rules

- Log send attempts and results
- Add timestamp + metadata
- TTL index for cleanup

# Frontend Development Guidelines

## React Structure

```
src/features/auth/  
src/features/profile/  
src/features/notifications/  
src/features/preferences/
```

## **UI Flow Requirements**

- Login / Signup screens with validation
- “Continue as Guest” entry
- Language and dietary preference UI
- Notification permission + subscription UI
- Token handling with React Query

## **Definition of Done**

- Secure login + guest login complete
- Refresh tokens working
- Profile updates saved and visible
- Notification delivery functional
- Clean UX and error handling in UI