# Railway WhatsApp Web Mikroservisi - Kapsamlı Deployment Rehberi

## Hızlı Başlangıç

### 1. Railway Hesap Kurulumu

1. [Railway.app](https://railway.app) sitesine gidin
2. “Login with GitHub” butonuna tıklayın
3. GitHub yetkilendirmesini tamamlayın

### 2. Proje Deployment - GitHub Repository Yöntemi (Önerilen)

#### Adım 2.1: GitHub Repository Oluşturma

1. GitHub’da yeni repository oluşturun (örn: whatsapp-microservice)
2. Bu klasörü repository’nize push edin:

cd whatsapp-railway-microservice  
git init  
git add .  
git commit -m "Initial WhatsApp microservice"  
git remote add origin https://github.com/KULLANICI\_ADI/whatsapp-microservice.git  
git push -u origin main

#### Adım 2.2: Railway’de Deploy

1. Railway dashboard’da “New Project” butonuna tıklayın
2. “Deploy from GitHub repo” seçeneğini seçin
3. Oluşturduğunuz repository’yi seçin
4. Railway otomatik olarak deployment başlatacak

### 3. Environment Variables Ayarlama

Railway project dashboard’da: 1. “Variables” sekmesine gidin 2. Aşağıdaki değişkenleri ekleyin:

NODE\_ENV=production  
PORT=3000  
ALLOWED\_ORIGINS=https://wln7xwqrttk2.space.minimax.io,https://xvxiwcbiqiqzfqisrvib.supabase.co

### 4. Deployment URL Alma

Deployment tamamlandığında: 1. “Deployments” sekmesinde successful deployment’ı göreceksiniz 2. “View Logs” ile deployment loglarını kontrol edin 3. “Domain” sekmesinde public URL’yi bulun 4. URL formatı: https://PROJE-ADI.up.railway.app

### 5. Health Check Test

Deployment başarılı olduğunu doğrulamak için:

curl https://PROJE-ADI.up.railway.app/health

Beklenen cevap:

{  
 "status": "healthy",  
 "timestamp": "2024-12-25T10:30:00.000Z",  
 "service": "whatsapp-web-microservice",  
 "version": "1.0.0",  
 "activeClients": 0  
}

## Supabase Integration

### Adım 1: Environment Variable Set Etme

1. [Supabase Dashboard](https://supabase.com/dashboard/project/xvxiwcbiqiqzfqisrvib) adresine gidin
2. Sol menüden “Settings” > “Environment Variables” seçin
3. “Add new variable” butonuna tıklayın
4. Değişken bilgilerini girin:
   * **Name**: WHATSAPP\_MICROSERVICE\_URL
   * **Value**: https://PROJE-ADI.up.railway.app (Railway’den aldığınız URL)
   * **Target**: Edge Functions seçin
5. “Save” butonuna tıklayın

### Adım 2: Edge Function Restart

Environment variable değişikliği sonrası Edge Function’ları restart edin: 1. Supabase Dashboard’da “Edge Functions” sekmesine gidin 2. whatsapp-production-service function’ını bulun 3. “…” menüsünden “Restart” seçin

## Test Süreci

### 1. Mikroservis Test

# Health check  
curl https://PROJE-ADI.up.railway.app/health  
  
# Session başlatma testi  
curl -X POST https://PROJE-ADI.up.railway.app/initialize-session \  
 -H "Content-Type: application/json" \  
 -H "X-User-ID: test\_user\_123"

### 2. Frontend Integration Test

1. https://wln7xwqrttk2.space.minimax.io adresine gidin
2. Test hesabı ile login: aagfpszd@minimax.com / AE62C8M6Gg
3. “WhatsApp Web” sayfasına gidin
4. “QR Kod Oluştur” butonuna tıklayın
5. Artık gerçek WhatsApp Web QR kodu göreceksiniz
6. WhatsApp mobile uygulamanız ile QR kodu tarayın

## Production Ready Features

### Güvenlik

* ✅ Rate limiting (100 request/15 dakika)
* ✅ CORS protection
* ✅ Helmet security headers
* ✅ Input validation
* ✅ User isolation per session

### Monitoring

* ✅ Health check endpoint
* ✅ Admin sessions endpoint
* ✅ Comprehensive logging
* ✅ Error tracking

### Performance

* ✅ Multi-user session support
* ✅ Memory efficient client management
* ✅ Graceful shutdown
* ✅ Auto-restart on failure

## Troubleshooting

### Common Issues

#### 1. Deployment Fails

* Railway logs’u kontrol edin
* Node.js versiyonu uyumluluğunu kontrol edin
* package.json syntax hatalarını kontrol edin

#### 2. QR Code Generation Slow

* Normal durum, 2-5 saniye sürebilir
* whatsapp-web.js kütüphanesinin Puppeteer başlatma süresi

#### 3. Session Connection Issues

* QR kod süresi (60 saniye) dolmuş olabilir
* WhatsApp mobile app versiyonunu kontrol edin
* Network bağlantısını kontrol edin

#### 4. Environment Variable Issues

* Supabase’de variable doğru set edildiğini kontrol edin
* Edge Function restart edildiğini kontrol edin
* Variable name’in tam olarak WHATSAPP\_MICROSERVICE\_URL olduğunu kontrol edin

### Log Monitoring

Railway dashboard’da: 1. “Deployments” sekmesine gidin 2. Son deployment’ı seçin 3. “View Logs” ile real-time logları izleyin

### Debug Commands

# Admin sessions listesi  
curl https://PROJE-ADI.up.railway.app/admin/sessions  
  
# Specific user session status  
curl -X POST https://PROJE-ADI.up.railway.app/session-status \  
 -H "X-User-ID: USER\_ID"

## Production Notları

### Resource Requirements

* Minimum RAM: 512MB
* Tavsiye edilen RAM: 1GB
* CPU: 0.5 vCPU yeterli
* Disk: 1GB

### Scaling

* Railway otomatik scaling sağlar
* Concurrent user sayısı RAM ile sınırlı
* Her WhatsApp session ~50-100MB RAM kullanır

### Backup & Recovery

* WhatsApp session data otomatik olarak persist edilir
* Railway automatic backup sağlar
* Session data .wwebjs\_auth klasöründe saklanır

## Support & Maintenance

### Regular Maintenance

* Monthly dependency updates önerili
* whatsapp-web.js kütüphanesini güncel tutun
* Railway platform güncellemelerini takip edin

### Monitoring Metrics

* Health check response time
* Active sessions count
* Error rate
* Memory usage

## Final Checklist

* ☐ Railway hesabı oluşturuldu
* ☐ GitHub repository oluşturuldu ve kod push edildi
* ☐ Railway’de project deploy edildi
* ☐ Environment variables set edildi
* ☐ Health check başarılı
* ☐ Supabase environment variable set edildi
* ☐ Edge function restart edildi
* ☐ Frontend test başarılı
* ☐ QR kod generation test başarılı
* ☐ WhatsApp mobile app ile connection test başarılı

Bu rehberi takip ederek tamamen functional, production-ready WhatsApp Web mikroservisi deploy edebilirsiniz.

## Sonuç

Bu mikroservis gerçek whatsapp-web.js kütüphanesi kullanarak authentic WhatsApp Web bağlantısı sağlar. Railway deployment sonrası kullanıcılar kendi WhatsApp hesaplarını platforma bağlayabilir ve gerçek mesajlaşma yapabilirler.