



# Workigom - Hızlı Başlangıç Kılavuzu

## Ön Gereksinimler

- Node.js (v18 veya üzeri)
- PostgreSQL (v14 veya üzeri)
- npm veya yarn

## 1. Projeyi İndirme

```
cd /home/ubuntu/workigom
```

## 2. Bağımlılıkların Yüklenmesi

### Backend

```
cd /home/ubuntu/workigom/backend  
npm install
```

### Frontend

```
cd /home/ubuntu/workigom  
npm install
```

## 3. Environment Variables

### Backend (.env)

```
cd /home/ubuntu/workigom/backend
```

.env dosyası zaten mevcut ve yapılandırılmış durumda:

```
PORT=3001  
DATABASE_URL=postgresql://user:password@localhost:5432/workigom  
JWT_SECRET=your-secret-key  
NODE_ENV=development
```

### Frontend (.env)

```
cd /home/ubuntu/workigom
```

.env dosyası zaten mevcut:

```
VITE_API_URL=http://localhost:3001/api
```

## 4. Database Kurulumu

```
cd /home/ubuntu/workigom/backend  
  
# Prisma client oluştur  
npm run prisma:generate  
  
# Database migration  
npm run prisma:migrate  
  
# Örnek veri yükle (opsiyonel)  
npm run prisma:seed
```

## 5. Uygulamayı Başlatma

### Terminal 1 - Backend

```
cd /home/ubuntu/workigom/backend  
npm run dev
```

Backend şu adreste çalışacak: **http://localhost:3001**

### Terminal 2 - Frontend

```
cd /home/ubuntu/workigom  
npm run dev
```

Frontend şu adreste çalışacak: **http://localhost:5173**

## 6. Uygulamaya Erişim

Tarayıcınızda şu adresi açın:

```
http://localhost:5173
```

## Test Kullanıcıları

Seed verisi ile oluşturulan test kullanıcıları:

### Şirket Hesapları

1. **Email:** company1@workigom.com  
**Şifre:** password123
2. **Email:** company2@workigom.com  
**Şifre:** password123

## Birey Hesapları

1. **Email:** user1@workigom.com

**Şifre:** password123

2. **Email:** user2@workigom.com

**Şifre:** password123

3. **Email:** user3@workigom.com

**Şifre:** password123

## API Test

API'nin çalışıp çalışmadığını test etmek için:

```
# Health check
curl http://localhost:3001/api/health

# İş ilanlarını listele
curl http://localhost:3001/api/jobs

# Bağışları listele
curl http://localhost:3001/api/donations
```

## Yaygın Sorunlar ve Çözümleri

### 1. Port Zaten Kullanımda

```
# Backend port (3001) kontrol
lsof -ti:3001 | xargs kill -9

# Frontend port (5173) kontrol
lsof -ti:5173 | xargs kill -9
```

### 2. Database Bağlantı Hatası

- PostgreSQL servisinin çalıştığından emin olun
- DATABASE\_URL'in doğru olduğundan emin olun
- Database'in oluşturulduğundan emin olun

### 3. TypeScript Hatası

```
# Backend
cd /home/ubuntu/workigom/backend
npm run build

# Frontend
cd /home/ubuntu/workigom
npm run build
```

## 4. Node Modules Sorunu

```
# Backend
cd /home/ubuntu/workigom/backend
rm -rf node_modules package-lock.json
npm install

# Frontend
cd /home/ubuntu/workigom
rm -rf node_modules package-lock.json
npm install
```

## Production Build

### Backend

```
cd /home/ubuntu/workigom/backend
npm run build
npm start
```

### Frontend

```
cd /home/ubuntu/workigom
npm run build
# build/ klasöründeki dosyalar serve edilebilir
```

## Faydalı Komutlar

### Database

```
cd /home/ubuntu/workigom/backend

# Prisma Studio (DB GUI)
npm run prisma:studio

# Yeni migration oluştur
npm run prisma:migrate

# Database sıfırla ve seed
npx prisma migrate reset
```

## Development

```
# Backend logs
cd /home/ubuntu/workigom/backend
npm run dev | grep -v "prisma:query"

# Frontend ile birlikte başlat (tek terminal)
cd /home/ubuntu/workigom/backend && npm run dev &
cd /home/ubuntu/workigom && npm run dev
```

## Proje Yapısı

```
workigom/
  └── backend/                               # Backend API
      ├── src/
      │   ├── controllers/                   # Route controllers
      │   ├── routes/                      # API routes
      │   ├── middleware/                  # Express middleware
      │   ├── utils/                       # Helper functions
      │   ├── config/                      # Configuration
      │   ├── types/                       # TypeScript types
      │   └── prisma/                     # Database schema & migrations
      └── uploads/                         # File uploads

  └── src/                                     # Frontend source
      ├── components/                    # React components
      ├── contexts/                     # Context providers
      ├── pages/                        # Page components
      ├── types/                         # TypeScript types
      └── utils/                         # Helper functions

  └── public/                                # Static assets
```

## Daha Fazla Bilgi

Detaylı dokümantasyon için:

- `DEVELOPMENT_STATUS.md` - Proje durumu ve detaylı bilgiler
- `backend/README.md` - Backend API dokümantasyonu
- `README.md` - Genel proje dokümantasyonu

**Not:** Tüm sistemler test edilmiş ve çalışır durumdadır! 🎉