

TPA Backend Deployment & Setup Documentation

1. Redis Setup

Check existing databases:

```
redis-cli
127.0.0.1:6379> INFO keyspace
```

Select DB:

```
127.0.0.1:6379> SELECT 1
OK
```

Environment variables (JSON style with comments):

```
{
  "REDIS_HOST": "127.0.0.1", // Redis server host
  "REDIS_PORT": "6379",      // Redis server port
  "REDIS_PASSWORD": "omer",  // Redis password
  "REDIS_URL": "redis://:omer@127.0.0.1:6379" // Full Redis URL
}
```

Export URL for shell:

```
export REDIS_URL=redis://:omer@127.0.0.1:6379
```

Node Redis client setup (`ioredis`):

```
import Redis from 'ioredis';

const redis = new Redis({
  port: parseInt(process.env.REDIS_PORT || '6379', 10),
  host: process.env.REDIS_HOST || '127.0.0.1',
  password: process.env.REDIS_PASSWORD,
});

redis.on('connect', () => console.log('Connected to Redis'));
redis.on('error', (err: Error) => console.error('Redis connection error:',
```

```
err));  
  
export default redis;
```

Reload Redis:

```
sudo systemctl restart redis
```

2. PostgreSQL Setup

Environment variables (`.env` or JSON style):

```
{  
  "DATABASE_NAME": "auth_db",  
  "DATABASE_USER": "omer",  
  "DATABASE_PASSWORD": "omerpas",  
  "DATABASE_HOST": "127.0.0.1",  
  "DATABASE_PORT": 5432  
}
```

Drizzle ORM Config (`drizzle.config.ts`):

```
import { defineConfig } from 'drizzle-kit';  
export default defineConfig({  
  out: './drizzle',  
  schema: './src/db/schema.ts',  
  dialect: 'postgresql',  
  dbCredentials: { url: process.env.DATABASE_URL! },  
});
```

Migrate Database:

```
bun run drizzle-kit migrate  
# Check applied migrations  
bun run drizzle-kit status
```

Create new migration:

```
bun run drizzle-kit create migration_name
```

3. Bun Scripts

Dev server with watch:

```
"scripts": {  
  "dev": "bun --watch run src/index.ts"  
}
```

Start server manually:

```
bun run start
```

Output should show:

```
Connected to Redis  
Up and running!
```

4. PM2 Process Management

Start App:

```
pm2 start src/index.ts --name TPA-Backend --watch --interpreter bun
```

Stop App:

```
pm2 stop TPA-Backend
```

Restart App:

```
pm2 restart TPA-Backend
```

Delete App:

```
pm2 delete TPA-Backend
```

View Logs & Status:

```
pm2 list  
pm2 logs TPA-Backend  
pm2 monit
```

Auto-start on server boot:

```
pm2 startup  
pm2 save
```

5. Accessing App via IP

- Ensure firewall allows port 3000 .
- Use http://<server-ip>:3000 in browser.
- If 404 appears, verify your routes exist.

This documentation consolidates all Redis/PostgreSQL/Bun/PM2 setup steps and commands for TPA Backend deployment.