

Deployment and Environment Configuration

1. Components

The platform deploys three services:

- API (`src/Helpdesk.Light.Api`)
- Worker (`src/Helpdesk.Light.Worker`)
- Web (`src/Helpdesk.Light.Web`, static Blazor WebAssembly)

2. Configuration Templates

Use these templates as production baselines:

- `deploy/appsettings.Api.Production.template.json`
- `deploy/appsettings.Worker.Production.template.json`
- `deploy/appsettings.Web.Production.template.json`

3. Secrets Strategy

Do not commit real secrets.

Inject secrets using your platform secret store or environment variables:

- `Jwt_SigningKey`
- `Ai_OpenAIApiKey`
- any real SMTP or mail adapter credentials (if not using console transport)

For local container deployment, copy `.env.example` to `.env` and set values.

4. Build Artifacts

Container build files are included:

- `docker/Dockerfile.api`
- `docker/Dockerfile.worker`
- `docker/Dockerfile.web`
- `docker-compose.yml`

Build images:

```
./scripts/build-images.sh local
```

5. Start in Development

```
dotnet restore Helpdesk.Light.slnx
dotnet build Helpdesk.Light.slnx -warnaserror
```

```
dotnet run --project src/Helpdesk.Light.Api  
dotnet run --project src/Helpdesk.Light.Worker  
dotnet run --project src/Helpdesk.Light.Web
```

6. Start with Docker Compose

```
cp .env.example .env  
# edit .env with secure values  
  
docker compose up --build
```

Endpoints:

- API: <http://localhost:8080>
- Web: <http://localhost:8082>

7. Startup Verification

After startup, verify:

- [GET /health/live](#)
- [GET /health/ready](#)
- admin login and seeded user auth
- ticket creation from web and email pathways
- worker dispatch loop running (check [GET /api/v1/ops/metrics](#) as admin)

8. Reproducibility Notes

- Containers build from pinned .NET 10 SDK/runtime images.
- Configuration is externalized with templates + environment variable overrides.
- Backup and restore scripts are in [scripts/](#) and operational procedures are in [docs/operations-runbook.md](#).