SMBItPro-Workshop

This repository now includes a .NET 10 implementation baseline for **Helpdesk-Light**.

Solution Layout

- `src/Helpdesk.Light.Domain`: domain entities and role constants.
- `src/Helpdesk.Light.Application`: contracts and service abstractions.
- `src/Helpdesk.Light.Infrastructure`: EF Core + SQLite, Identity entities, seed data, and tenant/customer services.
- `src/Helpdesk.Light.Api`: ASP.NET Core API with JWT auth, role-based access, customer/domain admin, and tenant resolution endpoints.
- `src/Helpdesk.Light.Web`: Blazor WebAssembly frontend shell and design system.
- `src/Helpdesk.Light.Worker`: worker service project scaffold.
- `tests/Helpdesk.Light.UnitTests`: domain and service unit tests.
- `tests/Helpdesk.Light.IntegrationTests`: API integration tests for auth, role policy, and tenant boundaries.

Prerequisites

- .NET SDK `10.0.103` (see `global.json`).

Restore, Build, and Test

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

CI-ready test command:

```
dotnet test Helpdesk.Light.slnx --configuration Release --no-build
```

CI/CD

GitHub Actions workflows are provided in `.github/workflows/`:

- `ci.yml`: runs on pull requests, pushes to branches (excluding `v*` tags), and manual dispatch.
  - Restores dependencies.
  - Builds in `Release` with warnings-as-errors.
  - Runs unit and integration tests with code coverage collection.
  - Uploads test result artifacts (`TRX` + coverage files).
- `release.yml`: runs on release tags (`v*`), release publication, or manual dispatch.
  - Packages API and Worker as self-contained binaries for:
    - `linux-x64`

- linux-arm64
- win-x64
- osx-x64
- osx-arm64
  - Packages Web as a static `wwwroot` bundle.
  - Generates `SHA256SUMS.txt`.
  - Attaches all packaged assets to the GitHub release.

To trigger a release package build from Git:

```
git tag v1.0.0
git push origin v1.0.0
```

Run API

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

Default API URL (from launch settings): `http://localhost:5283`

Seeded users:

- `admin@msp.local` / `Pass!12345` (`MspAdmin`)
- `tech@contoso.com` / `Pass!12345` (`Technician`, Contoso tenant)
- `tech@fabrikam.com` / `Pass!12345` (`Technician`, Fabrikam tenant)

Run Blazor WebAssembly UI

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

Default UI URL: `http://localhost:5167`.

The API base URL is configured in `src/Helpdesk.Light.Web/wwwroot/appsettings.json` via `ApiBaseUrl`.

Run Worker

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

Health, Analytics, and Operations Endpoints

- `GET /health/live` (anonymous)
- `GET /health/ready` (anonymous)
- `GET /api/v1/analytics/dashboard` (authenticated, tenant-scoped)

- `GET /api/v1/ops/metrics` (MSP admin)
- `GET /api/v1/ops/dead-letters` (MSP admin)
- `POST /api/v1/email/outbound/retry-dead-letter` (MSP admin)

KPI definitions are documented in `Helpdesk-Light/03-KPI-Definitions.md`.

Backup and Restore

- Backup: `./scripts/backup-helpdesk.sh`
- Restore: `./scripts/restore-helpdesk.sh`
- Runbook: `docs/operations-runbook.md`

Deployment

- Deployment docs: `docs/deployment.md`
- Config templates: `deploy/`
- Container assets: `docker/` + `docker-compose.yml`