

DomainTemplate CRUD Solution

One business object, implemented from PostgreSQL to reusable clients and management console.

Purpose

- Demonstrate a complete Spring Boot CRUD feature.
- Keep the public object separate from JPA persistence.
- Share one HTTP contract across Java, TypeScript, and Dart.
- Include a practical React console and bulk data workflows.

Runtime

- Spring Boot API under `/api/v1/domaintemplate`.
- PostgreSQL table `domain_templates` through Spring Data JPA.
- Java, TypeScript, and Dart clients.
- React, Vite, and MUI management console.
- MkDocs documentation and Marp presentation in the same repo.

Public Contract

Field	Rule
<code>id</code>	Server-generated <code>Long</code>
<code>code</code>	Required, unique, immutable string
<code>label</code>	Required string
<code>description</code>	Nullable string
Dates	Server-owned <code>Instant</code> values

DomainTemplate

```
{  
  "id": 42,  
  "code": "CUSTOMER",  
  "label": "Customer",  
  "description": "Customer master data",  
  "createdDate": "2026-08-30T17:00:00Z",  
  "lastModifiedDate": "2026-08-30T17:05:00Z"  
}
```

API

Method	Path
CRUD	<code>/api/v1/domaintemplate[/]{id}]</code>
<code>POST</code>	<code>/api/v1/domaintemplate/seed</code>
<code>POST</code>	<code>/api/v1/domaintemplate/import.{json,csv}</code>
<code>GET</code>	<code>/api/v1/domaintemplate/export.{json,csv}</code>

Seed and import upsert by `code` and ignore imported IDs and dates.

Model Boundary

- `DomainTemplate` : immutable public business object.
- `DomainTemplateEntity` : internal mutable JPA model.
- `domain_templates` : PostgreSQL persistence table.
- The service maps between public and persistence models.

Demo Database

`ddl-auto: update` initializes the current schema but does not drop the obsolete `domain_records` table.

```
docker compose -f docker/compose.yml down --volumes
make docker-up
```

Resetting the demo volume removes all stored data.

Deploy Quickly

```
make docker-up
```

- API: <http://localhost:8989>
- Swagger: <http://localhost:8989/swagger-ui.html>
- Console: <http://localhost:5173>
- Docs: <http://localhost:8000>

Take Away

- Presentation deck: <http://localhost:8082>
- PDF: [domaintemplate.pdf](#)
- PPTX: [domaintemplate.pptx](#)
- Presenter mode: press **p** on the deck.