

# MP.SPEC

---

MAPO SPEC: WebApp / Sito in versione dotNetCore 8 per la gestione del MES MAPO in particolare per le gestioni SPECiali (in attesa di finire di migrare SITE e ADM).

Comprende funzionalità amministrative avanzate e funzionalità standard.

## Sezioni Principali

Sono gestiti

- Articoli
- Operatori
- Assegnazione Operatori/Macchine a reparti
- PODL (Promesse ODL) prima della produzione
- ODL (Ordini di lavoro) legate ad effettive attività di produzione
- Gestione speciale dei KIT
- Gestione Dossier (es caso Baglietto)
- Gestione Ricette (tramite dossier)
- Gestione Parametri macchina
- Gestione giacenze magazzino (ove gestite)

Ci sono anche alcune pagine speciali di admin (ad esempio FluxLogStatus, usata per deduplicare i dati di FluxLog)

## Architettura

### Livello Dati (MP.Data)

Il layer dati centralizzato in **MP.Data** fornisce:

- **8 Repository**: Anag, Production, Dossier, FluxLog, System, MpVoc, MpMon, MpLand
- **Cache FusionCache** (Memory + Redis + DB) con invalidazione per tag
- **DI Registrations** attraverso **DataServiceCollectionExtensions** (**AddSpecDataLayer**, **AddLandDataLayer**, etc.)
- **MpDataService** come servizio singleton centrale per accesso a DB, Redis, MongoDB

### Livello Applicazione (MP.SPEC)

- **MpDataService** (Scoped, singleton in Program.cs): service wrapper sul DAL con caching FusionCache
- **Componenti Blazor Server**: layout interattivo server-side con **AddInteractiveServerComponents()**
- **API Controllers**: RecipeController, RecipeArchiveController per operazioni ricette
- **Autenticazione**: Windows Authentication (Negotiate) con Autorizzazione Blazor Server

### Livello Infrastruttura

- **SQL Server**: 4 DbContext (MoonProContext, MoonPro\_VocContext, MoonPro\_FluxContext, MoonPro\_STATSContext)

- **MongoDB**: storage ricette
- **Redis**: caching distribuito (FusionCache) + backplane
- **OpenTelemetry**: tracing su Uptrace (abilitabile via conf)
- **MessagePipe**: broadcasting messaggi real-time

## Refactoring Completati (Giugno 2026)

### Repository Pattern - Decomposizione MpSpecController

Il grande `MpSpecController/MpSpecRepository.cs` è stato scomposto in 8 repository specialistici:

#	Repository	Interfaccia	Metodi	DbContext
1	<b>Anag</b>	<code>IAnagRepository</code>	26	<code>MoonProContext</code>
2	<b>Production</b>	<code>IProductionRepository</code>	32	<code>MoonProContext</code>
3	<b>Dossier</b>	<code>IDossierRepository</code>	6	<code>MoonPro_FluxContext</code>
4	<b>FluxLog</b>	<code>IFluxLogRepository</code>	3	<code>MoonPro_FluxContext</code>
5	<b>System</b>	<code>ISystemRepository</code>	7	<code>MoonProContext + MoonProAdminContext</code>
6	<b>MpVoc</b>	<code>IMpVocRepository</code>	3	<code>MoonPro_VocContext</code>
7	<b>MpMon</b>	<code>IMpMonRepository</code>	4	<code>MoonProContext</code>
8	<b>MpLand</b>	<code>IMpLandRepository</code>	6	<code>MoonProContext</code>

Tutti i reference a `dbController.XXX()` nei servizi sono stati rimossi. I metodi originali rimangono nel file di repository come fallback documentato.

### Migrazione FusionCache

Tutti i metodi di lettura in `MpDataService.cs` sono stati migrati al pattern `GetOrFetchAsync<T>()`:

- **L1 MemoryCache**: 1/3 della scadenza totale
- **L2 Redis (Distributed)**: TTL configurabile
- **L3 Database**: fetch diretto dal DbContext
- **Invalidazione**: per tag (es. `Utils.redisArtList`, `Utils.redisOdlByKey`)
- **48+ metodi** migrati o confermati corretti

### Fix DI e Static State (MP.AppAuth)

Risolto il null reference error originario (Cannot provide a value for property 'AAService' on type 'CmpTop'):

- Rimpiazzati tutti i `static IConfiguration _configuration` con `readonly` istanza nei controllers
- `AppAuthService` ora riceve i controllers via constructor DI invece di crearli con `new()`
- RegISTRAZIONI DI centralizzate in `DataServiceCollectionExtensions.cs`
- Tutti i controllers e services `mp.appauth` registrati come Scoped

### Build

Tutte le 10 soluzioni compilano con successo (0 errori):

MP.SPEC, MP.Data, MP.Land, MP.MON, MP.TAB3, MP.Stats, MP.INVE, MP.IOC, MP.RIOC, MP.Prog, lobConf