

DateValidationServiceService

August 17, 2026

DateValidationServiceService

Kind: Class

Source: Pams/Web/Pams.WebApp/Client/app/services/sales/date-validation-service.service.ts
(line 15)

Part of: [Pams](#)

`DateValidationServiceService` coordinates date validation and mandatory-field checks in the sales client. It triggers date validation, registers validators, and applies mandatory rules for general fields, entities, and types.

Methods

Method	Signature	Returns
<code>FireDateValidation</code>	<code>FireDateValidation(data: number)</code>	<code>void</code>
<code>AddValidator</code>	<code>AddValidator(type: number, Validator: validator)</code>	<code>void</code>
<code>validateMandatory</code>	<code>validateMandatory(type: number)</code>	<code>void</code>
<code>validateMandatoryByEntity</code>	<code>validateMandatoryByEntity(type: number, inquiry: Inquiry)</code>	<code>void</code>
<code>validateMandatorybyType</code>	<code>validateMandatorybyType(type: number)</code>	<code>void</code>

Properties

Property	Type
<code>validation\$</code>	<code>any</code>

Where it refuses work

- `DateValidationServiceService` stops the work with an early return when `type < 10`.

Diagram

```
graph LR
    SalesForm[Sales UI] --> Service[DateValidationServiceService]
    Service --> AddValidator[AddValidator]
    Service --> FireDateValidation[FireDateValidation]
    Service --> Mandatory[validateMandatory]
    Mandatory --> ByEntity[validateMandatoryByEntity]
    Mandatory --> ByType[validateMandatorybyType]
```

Usage

```
import { Component, inject } from '@angular/core';
import { DateValidationServiceService } from '../services/sales/date-validation-service.service';

@Component({
  selector: 'app-sales-entry',
  template: '<button (click)="validateDates()">Validate dates</button>',
})
export class SalesEntryComponent {
  private readonly dateValidationService = inject(DateValidationServiceService);

  validateDates(): void {
    this.dateValidationService.FireDateValidation();
  }
}
```

AI Coding Instructions

- Call `FireDateValidation()` when sales form date values change or before submitting date-dependent data.
- Register validation behavior through `AddValidator()` rather than duplicating date checks in components.
- Keep mandatory validation rules aligned with `validateMandatoryByEntity()` and `validateMandatorybyType()` scopes.
- Preserve the existing method names, including their casing, when adding callers or tests.

Used by

1 reference from 1 file. Each is a place in this repository where the symbol is actually used — go read one rather than trusting an example.

Imported by (1)

- `SharedServicesModule` — `Pams/Web/Pams.WebApp/Client/app/services/shared-service.module.ts :1`