

MrmAccountPayableComponent

August 18, 2026

MrmAccountPayableComponent

Kind: Class

Source: Pams/Web/Pams.Web/Client/app/components/dashboard/dashboard-library/widgets/mrm-account-payable/mrm-account-payable.component.ts (line 74)

Part of: [Pams](#)

MrmAccountPayableComponent is an Angular dashboard widget for displaying accounts payable data. It retrieves currency and authorization information, loads and saves dashboard data, and calculates payable totals by year and month.

Methods

Method	Signature	Returns
getCurrencies	getCurrencies()	void
GetAuthorizationAccess	GetAuthorizationAccess(url: string)	void
ngOnInit	ngOnInit()	void
ngOnChanges	ngOnChanges()	void
ngOnDestroy	ngOnDestroy()	void
getDashboardData	getDashboardData()	void
initializingData	initializingData(value: BranchData)	void
saveDashboardData	saveDashboardData()	void
calculateSumByYear	calculateSumByYear()	void
calculateSumByMonth	calculateSumByMonth(monthIndex: number)	void
AddNewCurrency	AddNewCurrency(account: number, SalesCom: number)	void
onCurrencyChanged	onCurrencyChanged(e: undefined, account: number, SalesCom: number, currencyIndex: number)	void

Method	Signature	Returns
	number)	
translateMonthName	translateMonthName(MonthId: number)	string
getpriceFormatted	getpriceFormatted(x: undefined)	string
onResize	onResize()	void
setChaching	setChaching()	void
getChaching	getChaching()	void
switchByGroupsHandler	switchByGroupsHandler()	void
switchBasedOnTendencyHandler	switchBasedOnTendencyHandler()	void
comparingWithPrevMonth	comparingWithPrevMonth(month: MonthData, accountIndex: number, salesComIndex: number, currencyIndex: number, monthIndex: number, propertyName: string)	string
comparingSummeryWithPrevMonth	comparingSummeryWithPrevMonth(month: MonthData, monthIndex: number, propertyName: string)	string
isCurrentMonth	isCurrentMonth(month: MonthData)	boolean
getpriceForecastAmount	getpriceForecastAmount(month: MonthData)	number
expectedCurrencyChange	expectedCurrencyChange(e: undefined)	void
selectText	selectText(e: undefined)	void
editSelectedMonthPrepare	editSelectedMonthPrepare(month: MonthData, supplierId: number)	void
mouseoverRowSpan	mouseoverRowSpan(e: undefined, bIndex: number)	void
mouseoutRowSpan	mouseoutRowSpan(e: undefined, bIndex: number)	void
openPopup	openPopup(n: string, t: string, i: string, btnTxt: string)	void
savePopup	savePopup()	void
ClosewithDonotSave	ClosewithDonotSave()	void
OnOpenedPopup	OnOpenedPopup()	void
closePopup	closePopup()	void

Properties

Property	Type
sharedPopup	SharedPopupComponent
_widget	Widget
_widgetType	string
withoutOptionBar	boolean
editMode	boolean
bookingOrSales	number
byGroups	boolean
selectedTeam	number
forecastRuleId	number
data	BranchData
allowedtoEdit	boolean
allowedtoEnableEdting	boolean
widgetEditRequest	any
widgetEditEnd	any
forecastRules	any[]
currencies	CompanyCurrency[]
selectedYear	number
currentMonthIndex	number
numYearsDatasource	any[]
todayWidgetsEnum	any
cachingWidgetData	CachingWidgetData
originalData	BranchData
selectedData	BranchData
singleMonths	MonthData
monthsSummery	MonthsSum[]
nonGroupAccountReceivables	MonthData[]
branchCurrency	any

Property	Type
basedOnTendency	boolean
subscription	Subscription
SeriseRange	any[]
echartsIntance	any
totalTargets	number
yearBooking	number
totalForecasts	number
tempTotalTarget	TotalTargets
totalProfitAndLossAmount	number
totalOthers	number
totalTotalSGAndA	number
totalFORX	number
totalTotal	number
totalCumulative	number
totalPercentageUsedBudget	number
popup	any
searchText	string

Where it refuses work

- `MrmAccountPayableComponent` stops the work with an early return when `MonthId == 12` .
- `MrmAccountPayableComponent` stops the work with an early return when `month[propertyName] < this.data.Products.Sales[currencyIndex].Details[monthIndex - 1][pro...` .
- `MrmAccountPayableComponent` stops the work with an early return when `month[propertyName] < this.data.Scrap.Sales[currencyIndex].Details[monthIndex - 1][proper...` .
- `MrmAccountPayableComponent` stops the work with an early return when `month[propertyName] < this.monthsSummery[monthIndex - 1][propertyName]` .
- `MrmAccountPayableComponent` stops the work with an early return when `this.allowedtoEnableEdting` .

- `MrmAccountPayableComponent` stops the work with an early return when `this.allowedtoEdit` .

Diagram

```
graph LR
    A[Angular Lifecycle] --> B[MrmAccountPayableComponent]
    B --> C[getCurrencies]
    B --> D[GetAuthorizationAccess]
    B --> E[getDashboardData]
    B --> F[initializingData]
    B --> G[saveDashboardData]
    E --> H[calculateSumByYear]
    E --> I[calculateSumByMonth]
    B --> J[ngOnDestroy]
```

Usage

```
import { ComponentFixture, TestBed } from '@angular/core/testing';
import { MrmAccountPayableComponent } from './mrm-account-payable.component';

describe('MrmAccountPayableComponent', () => {
  let fixture: ComponentFixture<MrmAccountPayableComponent>;
  let component: MrmAccountPayableComponent;

  beforeEach(() => {
    fixture = TestBed.createComponent(MrmAccountPayableComponent);
    component = fixture.componentInstance;

    fixture.detectChanges();
  });

  it('loads payable dashboard data', () => {
    component.GetAuthorizationAccess();
    component.getCurrencies();
    component.getDashboardData();

    expect(component).toBeTruthy();
  });
});
```

AI Coding Instructions

- Keep data loading inside the existing initialization and lifecycle flow rather than calling dashboard APIs from templates.

- Call `GetAuthorizationAccess()` before exposing or saving payable dashboard data that depends on user permissions.
- Keep currency retrieval through `getCurrencies()` so payable values use the component's configured currency data.
- Update both `calculateSumByYear()` and `calculateSumByMonth()` when changing the payable data structure or aggregation rules.
- Release subscriptions, timers, or other resources in `ngOnDestroy()` when adding asynchronous behavior.

Used by

3 references from 3 files. Each is a place in this repository where the symbol is actually used — go read one rather than trusting an example.

Imported by (3)

- `TranslateHandler` — `Pams/Web/Pams.Web/Client/app/app.module.ts :1`
- `DashboardTabComponent` — `Pams/Web/Pams.Web/Client/app/components/dashboard/dashboard-library/dashboard-tab/dashboard-tab.component.ts :1`
- `ReportsTabComponent` — `Pams/Web/Pams.Web/Client/app/components/dashboard/reports-library/reports-tab/reports-tab.component.ts :1`