WidgetForecastBacklogPaymentComponent

August 18, 2026

WidgetForecastBacklogPaymentComponent

Kind: Class

Source: `Pams/Web/Pams.Web/Client/app/components/dashboard/dashboard-libary/widgets/widget-forecast-backlog-payment/widget-forecast-backlog-payment.component.ts` (line 26)

Part of: Pams

`WidgetForecastBacklogPaymentComponent` manages forecast backlog payment data for a dashboard widget. It loads dashboard and My Disk data, processes responses, handles sales pipeline chart interactions, and controls invoice display settings.

Implements: `OnInit`

Methods

Method	Signature	Returns
`getDashboardData`	`getDashboardData()`	`void`
`getMyDiskData`	`getMyDiskData()`	`void`
`afterResponse`	`afterResponse(Data: WidgetBacklogInvoices[])`	`void`
`borderColor`	`borderColor(stage: string)`	`string`
`ngOnInit`	`ngOnInit()`	`void`
`onChartInit`	`onChartInit(e: any)`	`void`
`onSalesPipelineChartClick`	`onSalesPipelineChartClick(e: undefined)`	`void`
`onSalesPipelineChartInMyDiskTabClick`	`onSalesPipelineChartInMyDiskTabClick(e: undefined)`	`void`
`switchCommissionInvoicesOff`	`switchCommissionInvoicesOff()`	`void`
`switchSalesInvoicesOff`	`switchSalesInvoicesOff()`	`void`

Method	Signature	Returns
`ngOnDestroy`	`ngOnDestroy()`	`void`

Properties

Property	Type
`_widget`	`Widget`
`_myDiskFilterObj`	`MyDiskFilterObj`
`ChartInMyDiskTabClick`	`any`
`todayWidgetsEnum`	`any`
`subscription`	`Subscription`
`maxValueForYaxis`	`number`
`initOpts`	`any`
`series`	`any[]`
`headerData`	`any[]`
`data`	`any[]`
`datasource`	`WidgetBacklogInvoices[]`
`originalDatasource`	`WidgetBacklogInvoices[]`
`echartsIntance`	`any`
`CommissionInvoices`	`boolean`
`SalesInvoices`	`boolean`
`isLoading`	`boolean`
`options`	`any`

Where it refuses work

- `WidgetForecastBacklogPaymentComponent` stops the work with an early return when `stage == "Overdue"`.

Diagram

```
graph LR
  Init[ngOnInit] --> Dashboard[getDashboardData]
```

```
  Init --> MyDisk[getMyDiskData]
  Dashboard --> Response[afterResponse]
  MyDisk --> Response
  Response --> Chart[onChartInit]
  Chart --> PipelineClick[onSalesPipelineChartClick]
  Chart --> MyDiskClick[onSalesPipelineChartInMyDiskTabClick]
  Response --> Border[borderColor]
  InvoiceControls[switchCommissionInvoicesOff / switchSalesInvoicesOff] --> Response
```

Usage

```
import { ComponentFixture, TestBed } from '@angular/core/testing';
import { WidgetForecastBacklogPaymentComponent } from './widget-forecast-backlog-payment.compon

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

  beforeEach(async () => {
    await TestBed.configureTestingModule({
      declarations: [WidgetForecastBacklogPaymentComponent],
    }).compileComponents();

    fixture = TestBed.createComponent(WidgetForecastBacklogPaymentComponent);
    component = fixture.componentInstance;

    fixture.detectChanges();
  });

  it('loads widget data and supports invoice controls', () => {
    component.getDashboardData();
    component.getMyDiskData();

    component.switchCommissionInvoicesOff();
    component.switchSalesInvoicesOff();

    expect(component.borderColor()).toBeDefined();
  });
});
```

AI Coding Instructions

- Keep data loading in `getDashboardData()` and `getMyDiskData()` so dashboard and My Disk flows remain separate.
- Process API results through `afterResponse()` before updating chart-related state.

- Initialize chart behavior from `onChartInit()` and route chart click events to the matching sales pipeline handler.
- Preserve the separate handlers for dashboard and My Disk sales pipeline clicks, since they may require different navigation or filtering behavior.
- Update invoice visibility through `switchCommissionInvoicesOff()` and `switchSalesInvoicesOff()` rather than changing related state directly.

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)

- `TranslateHandler` — `Pams/Web/Pams.Web/Client/app/app.module.ts`:1