Project Title: Sales Performance Reporting

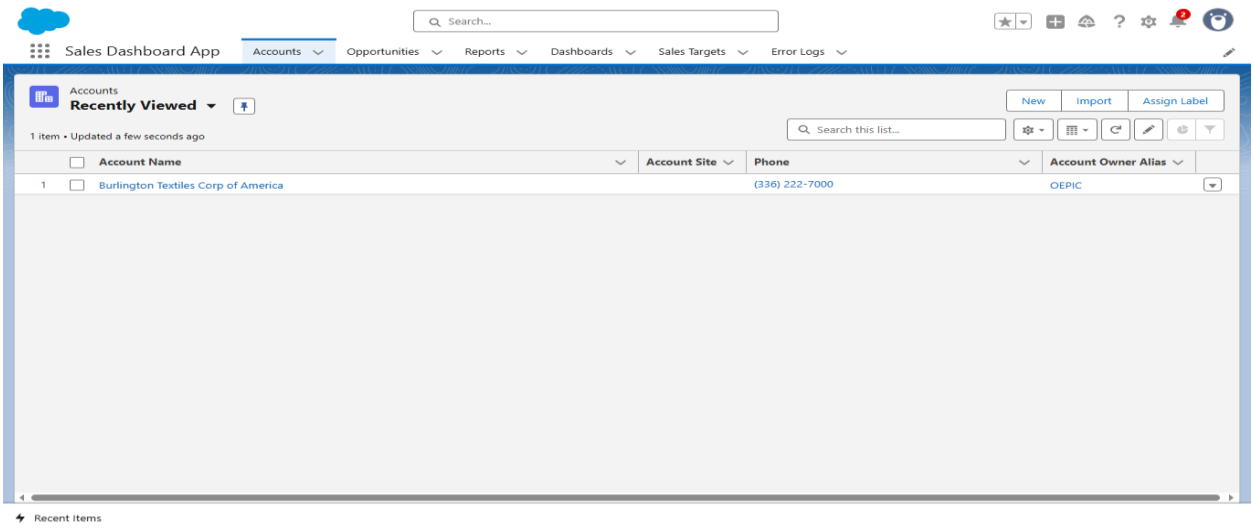
Phase 6: User Interface Development (with Lightning App Builder)

Objective:
In this phase, I designed and customized the **Sales Performance Reporting interface** using **Salesforce Lightning App Builder**. The goal was to make the UI simple, interactive, and role-based so that Sales Reps, Managers, and Executives can easily track performance through dashboards, reports, and custom components — all built and configured visually with the App Builder.

---

1. Lightning App Builder

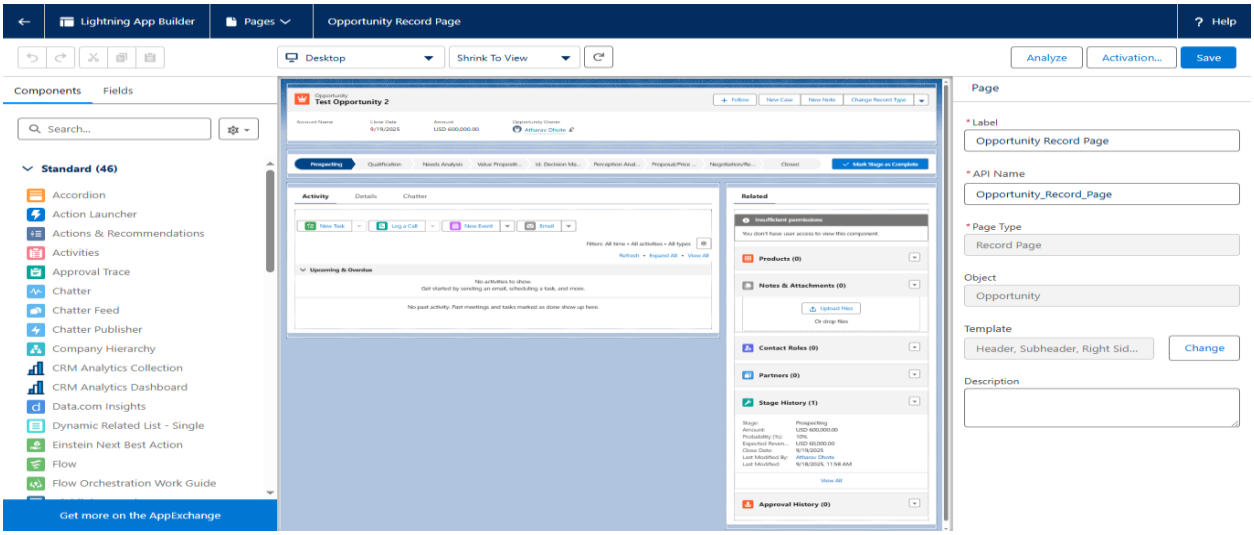
- I created a new **custom Lightning app** named **"Sales Dashboard App"** directly in Lightning App Builder.
- I added navigation items: **Accounts, Opportunities, Reports, Dashboards, Sales Targets (custom object), and Error Logs (custom object)**.
- I also applied simple branding like app name and icon to give it a professional look.

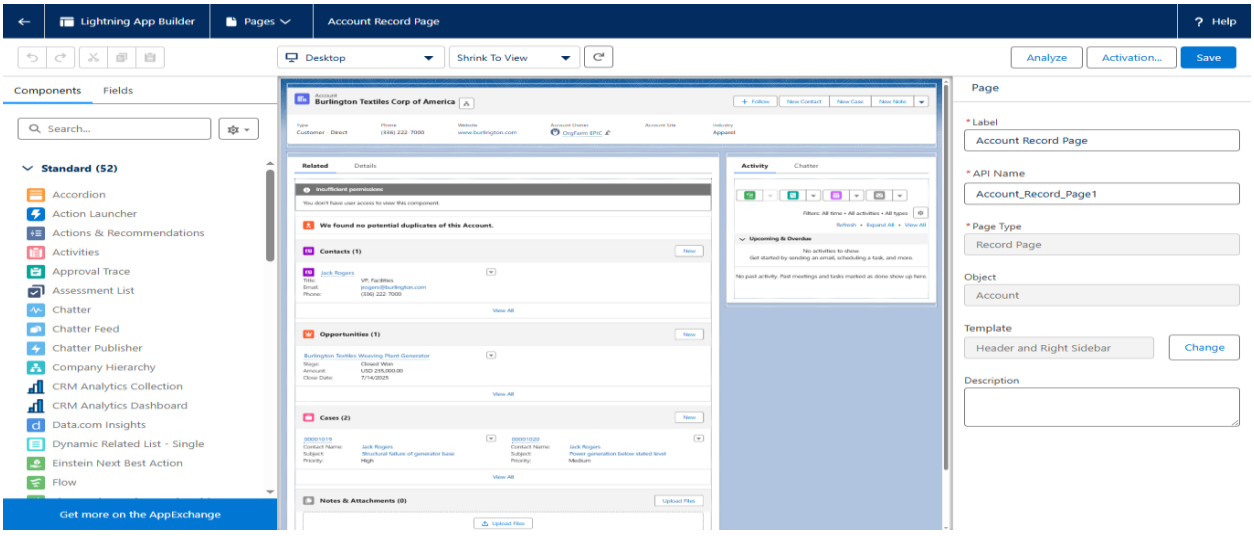


---

2. Record Pages

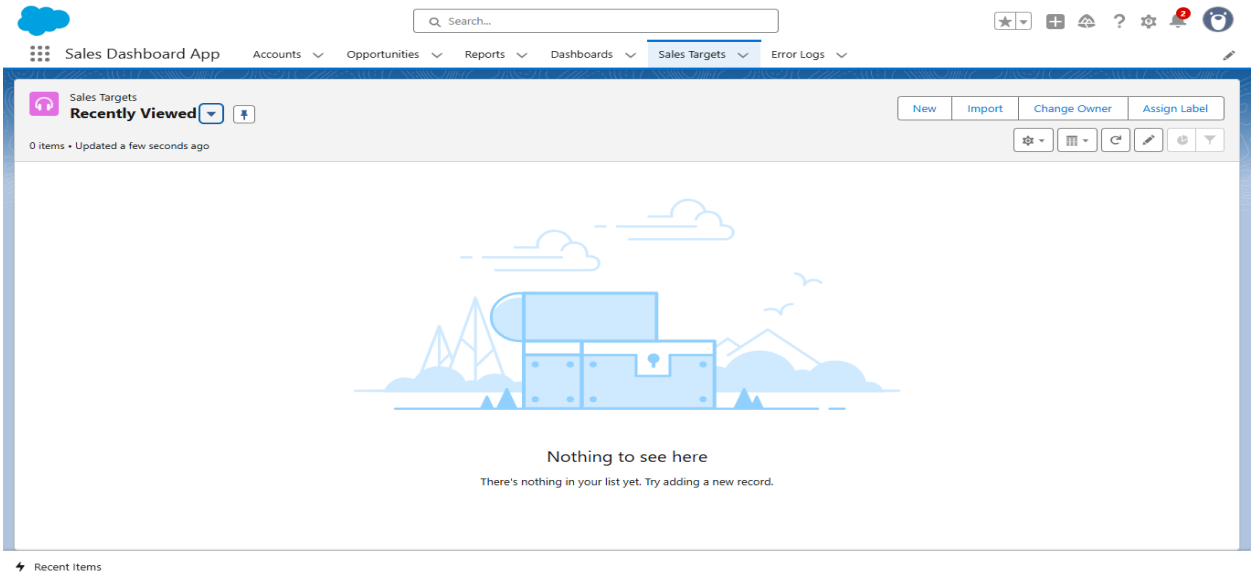
- I opened Opportunity and Account record pages in **Lightning App Builder** and customized them:
    - **Opportunity Page:** Added a **Highlights Panel** for key fields, **Report Chart** for pipeline, and **Related Lists** for activities.
    - **Account Page:** Added a **Report Chart** showing Opportunities by Region and a related list of Opportunities.





3. Home Page Layouts

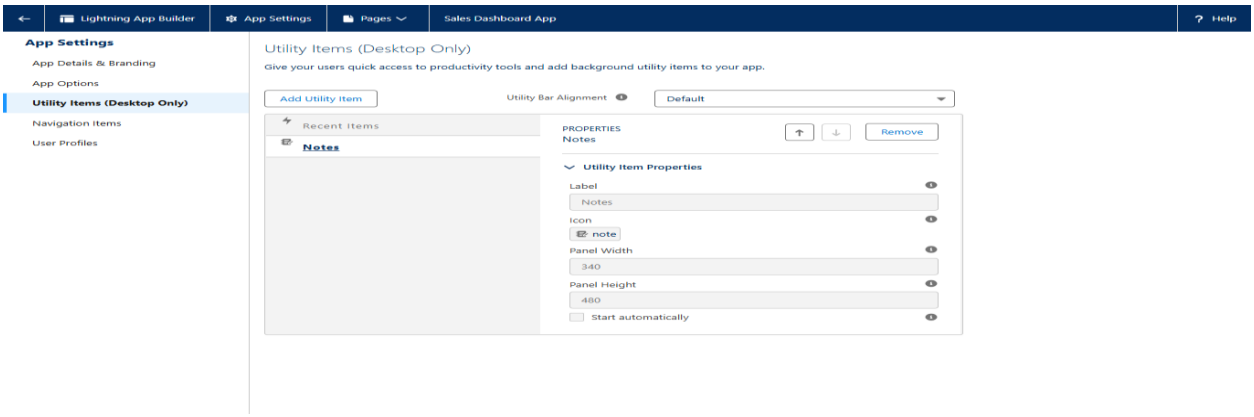
- I customized the **Home Page** with App Builder for two roles:
    - **Sales Reps:** Tasks, Notifications, Open Opportunities.
    - **Managers:** Dashboard with Quota vs Achievement and High-Value Deals.



---

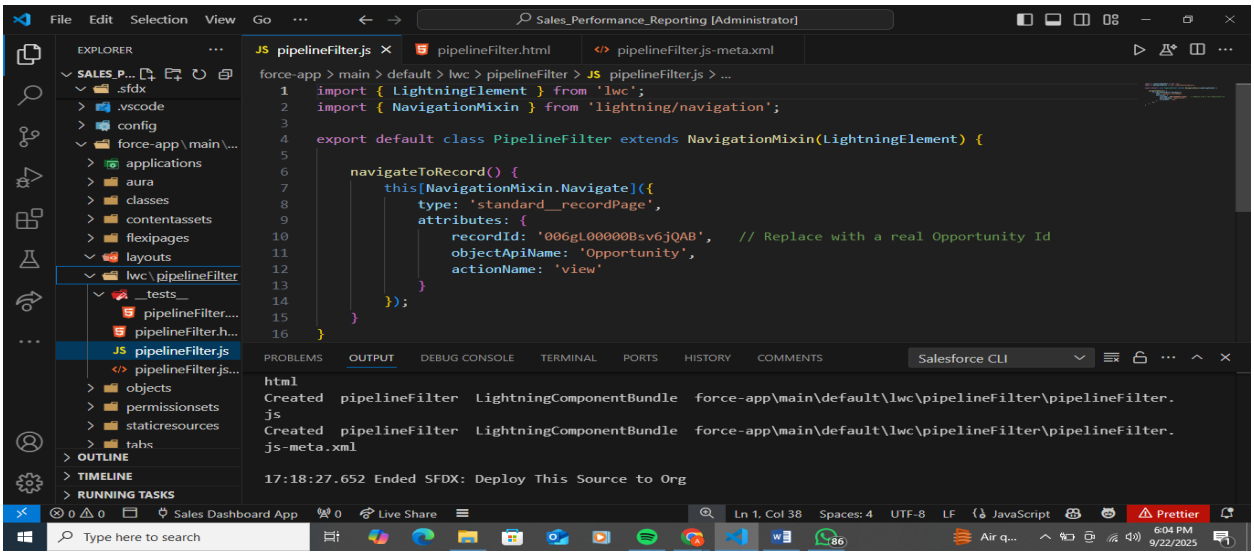
4. Utility Bar

- In App Builder, I configured a **Utility Bar** with:
    - **Notes** for quick meeting notes.
    - **Recent Items**



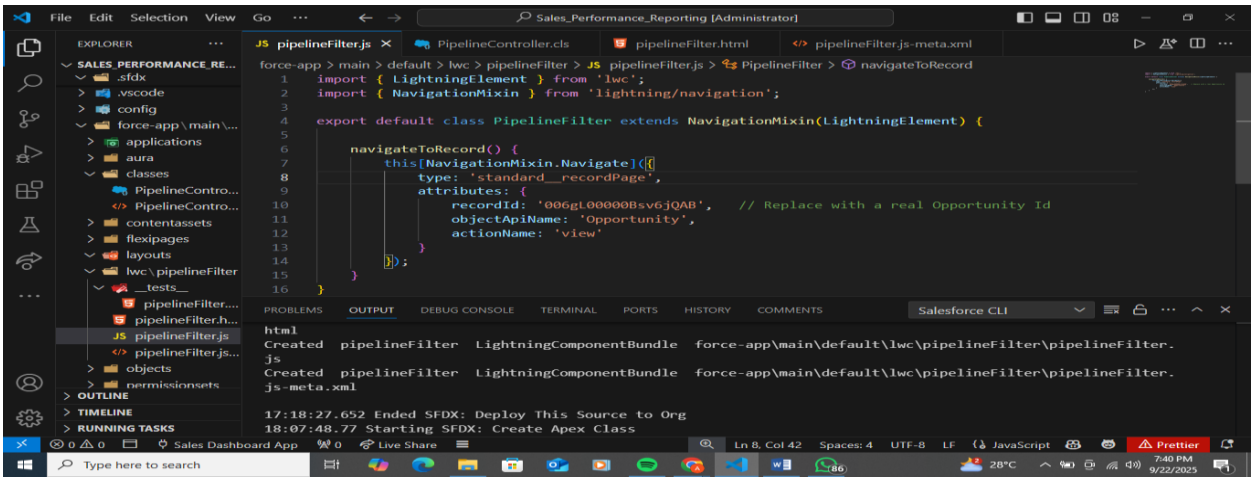
5. Lightning Web Components (LWC)

- I deployed LWCs and then **dragged them into the App Builder page layout**:
    - **PipelineFilterLWC** → filters Opportunities by region and stage.



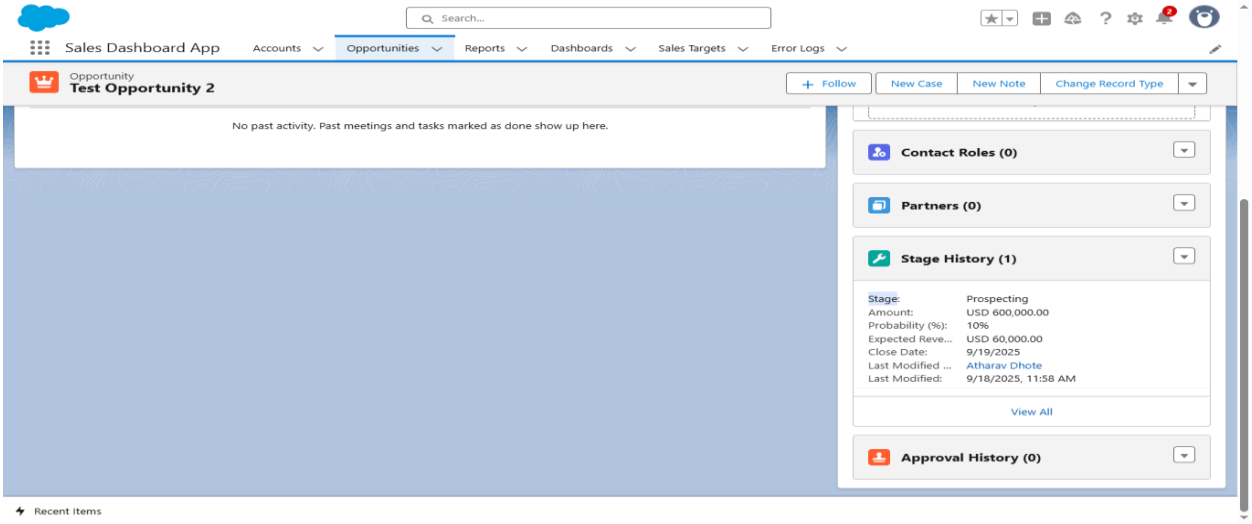
6. Apex with LWC

- I connected LWCs with Apex classes to fetch live data, and then placed those LWCs on the dashboard page in App Builder.
    - Example: **getOpportunitiesByRegion** → feeds data to the pipeline chart.



7. Wire Adapters

- LWCs used @wire adapters with Apex and record data, and I exposed those LWCs on the App Builder canvas.
- Example: The Opportunity stage chart automatically updates when data changes.



---

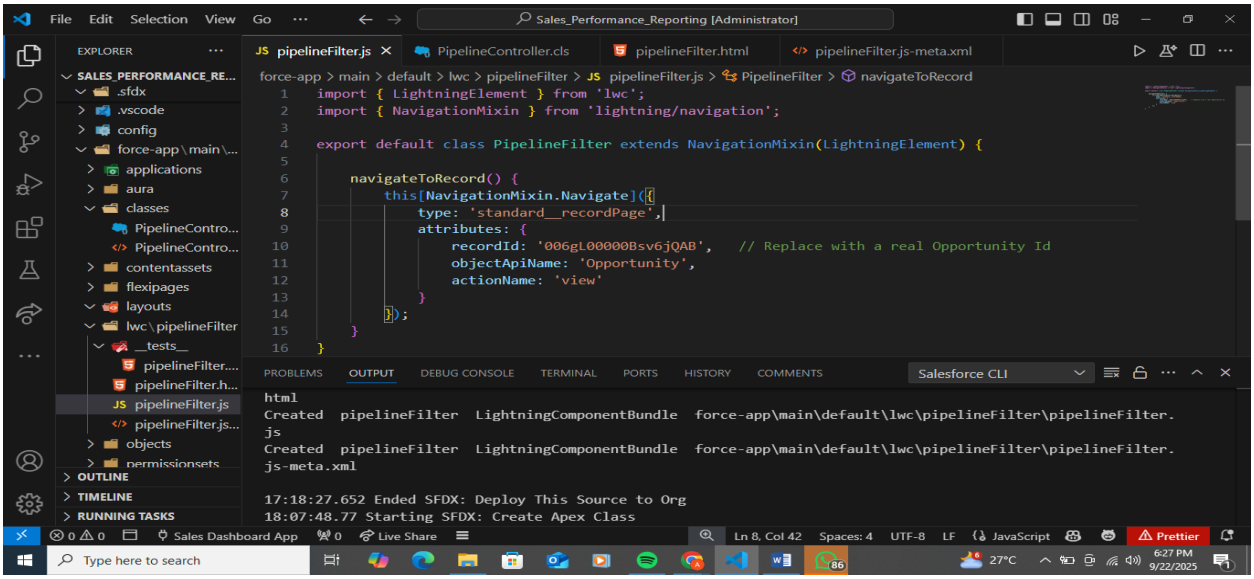
8. Imperative Apex Calls

- I built LWCs that fetch filtered data on button clicks (using imperative calls).
- Added these LWCs into App Builder so users can use interactive filters.

---

9. Navigation Service

- In LWCs placed on pages via App Builder, I used **Navigation Service** so that:
  - Clicking an Opportunity in a chart opens its detail page.
  - Managers can jump from charts to related dashboards.


File Edit Selection View Go
Sales_Performance_Reporting [Administrator]
EXPLORER
SALES_PERFORMANCE_RE...
.sfdx
.vscode
config
force-app\main\...
applications
aura
classes
PipelineContro...
PipelineContro...
contentassets
flexipages
layouts
lwc\pipelineFilter
__tests__
pipelineFilter....
pipelineFilter.h...
pipelineFilter.js
pipelineFilter.js...
objects
permissionsets
OUTLINE
TIMELINE
RUNNING TASKS
pipelineFilter.js
PipelineController.cls
pipelineFilter.html
pipelineFilter.js-meta.xml
force-app > main > default > lwc > pipelineFilter > pipelineFilter.js > PipelineFilter > navigateToRecord
1 import { LightningElement } from 'lwc';
2 import { NavigationMixin } from 'lightning/navigation';
3
4 export default class PipelineFilter extends NavigationMixin(LightningElement) {
5
6 navigateToRecord() {
7 this[NavigationMixin.Navigate]({
8 type: 'standard__recordPage',
9 attributes: {
10 recordId: '006gL00000Bsv6jQAB', // Replace with a real Opportunity Id
11 objectApiName: 'Opportunity',
12 actionName: 'view'
13 }
14 });
15 }
16 }
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS HISTORY COMMENTS
Salesforce CLI
html
Created pipelineFilter LightningComponentBundle force-app\main\default\lwc\pipelineFilter\pipelineFilter.
js
Created pipelineFilter LightningComponentBundle force-app\main\default\lwc\pipelineFilter\pipelineFilter.
js-meta.xml
17:18:27.652 Ended SFDX: Deploy This Source to Org
18:07:48.77 Starting SFDX: Create Apex Class
Sales Dashboard App
Live Share
Ln 8, Col 42 Spaces: 4 UTF-8 LF JavaScript
Prettier
Type here to search
27°C
6:27 PM
9/22/2025