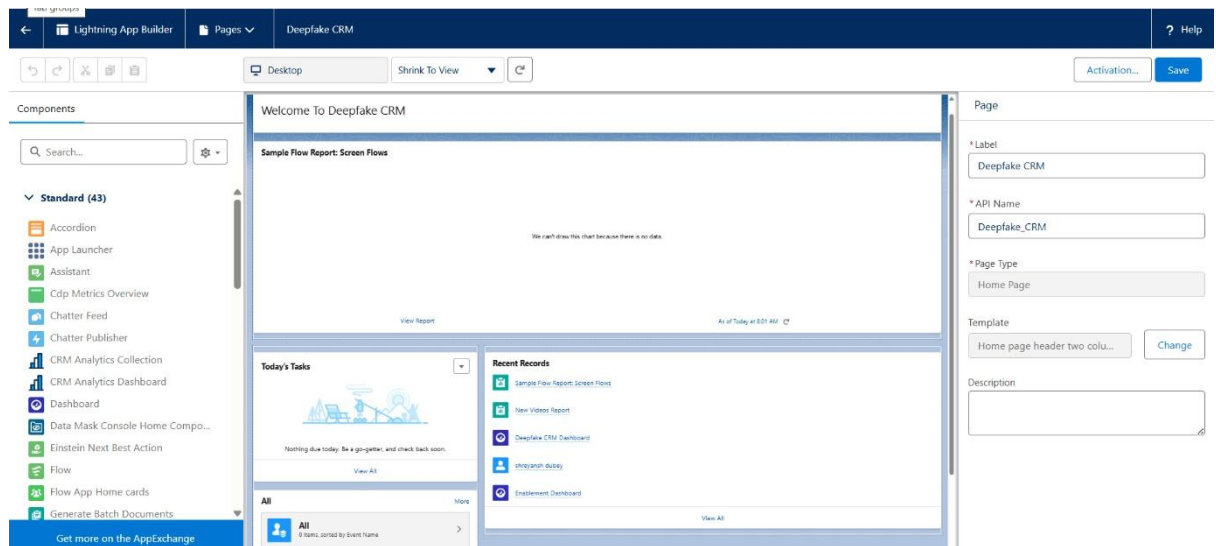

Phase 6: User Interface Development – Salesforce

1. Lightning App Builder

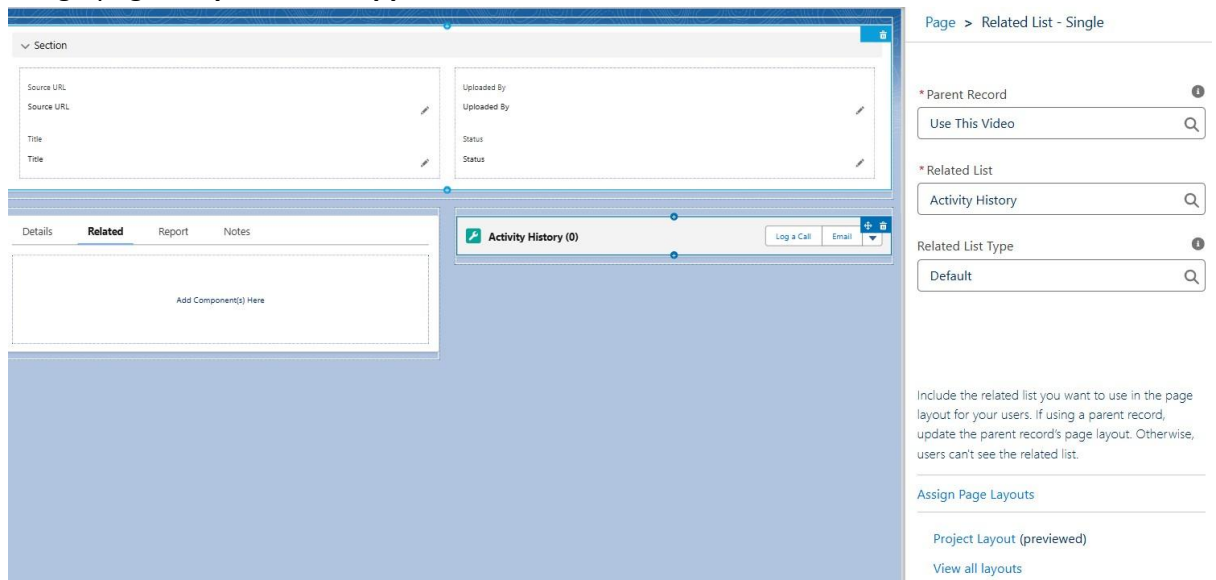
- Drag-and-drop tool to create **App, Home, or Record** pages.
- Types of pages:
 - **App Page:** Standalone page in App Launcher.
 - **Home Page:** Landing page with dashboards, reports, and components.
 - **Record Page:** Custom layout for specific objects (e.g., Video).
- Features:
 - Dynamic visibility
 - Tabs for organization
 - Custom LWC integration
- **Tip:** Use tabs template for Record Pages to organize Details, Related Lists, and custom components.



2. Record Pages

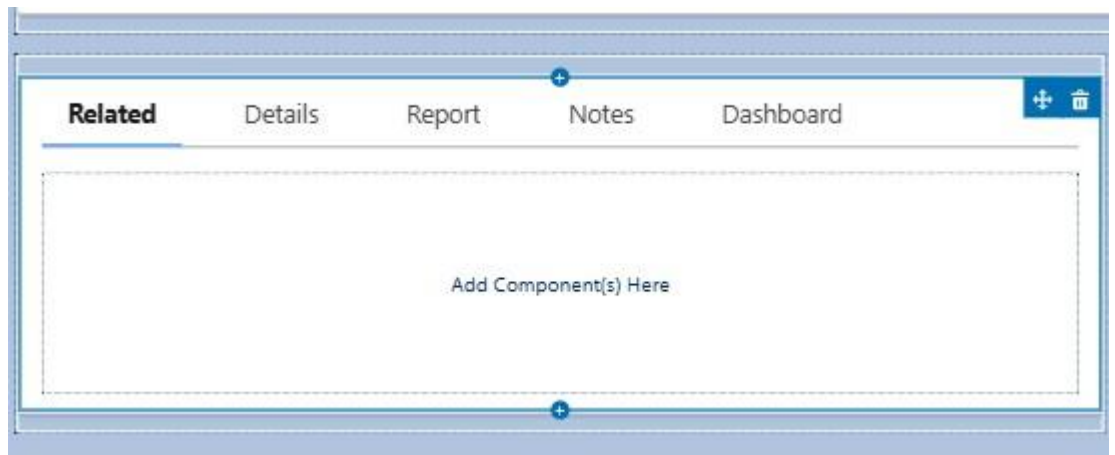
- Customizes layout for **object records**.

- Include:
 - **Highlights Panel:** Key fields at top.
 - **Tabs:** Organize Details, Related Lists, and Custom LWCs.
 - **Custom Components:** Video Player, Alerts, Detection status.
- Assign pages to **profiles or apps**.



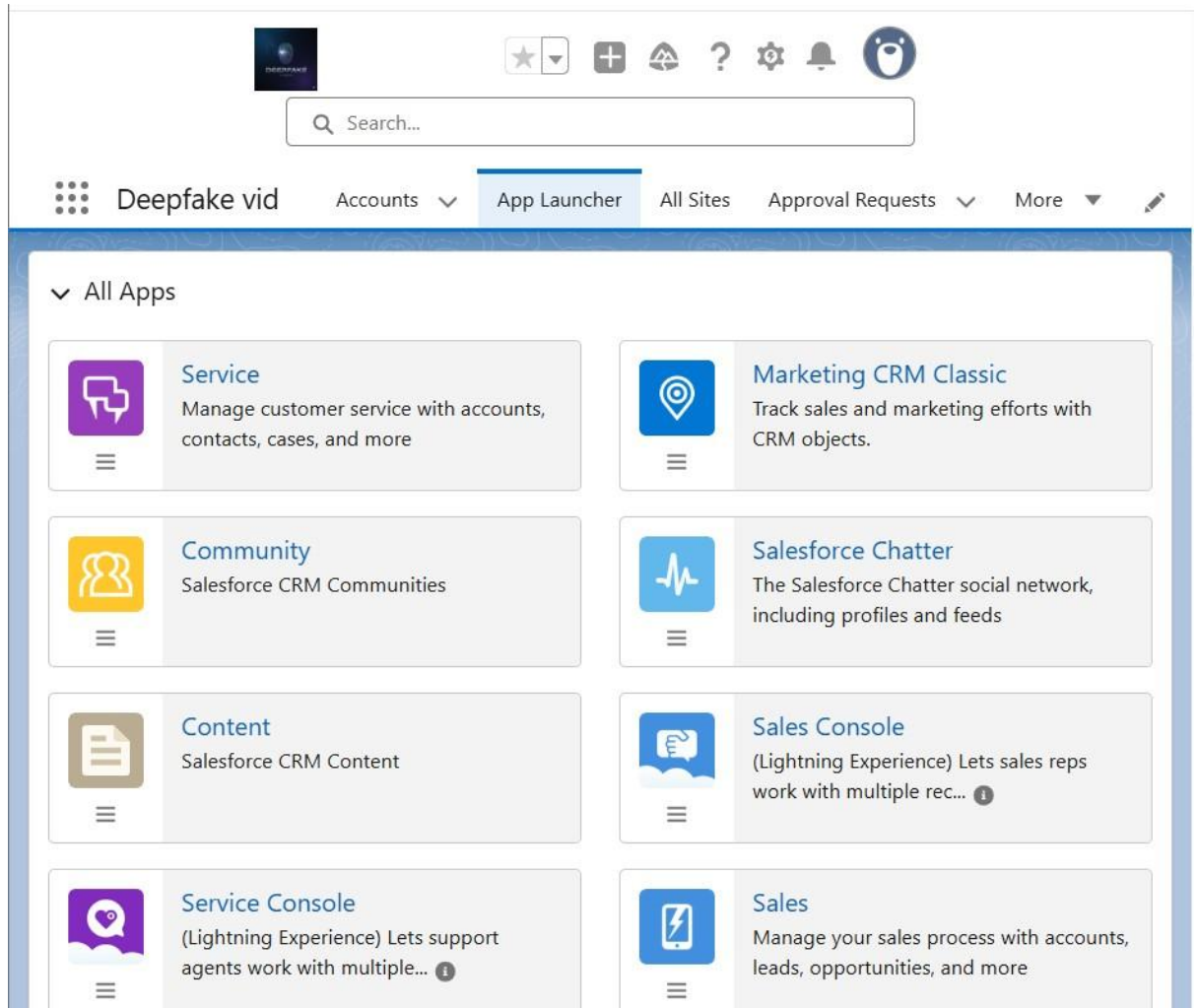
3. Tabs

- Organize information efficiently.
- Examples for Video object:
 - Details (Record Fields) ○ Related
 - (Alerts, Comments) ○ Video Player (LWC) ○
 - Detection/Info (Custom LWC or Rich Text) ○
 - Report ○ Notes ○ Dashboard



4. Home Page Layouts

- Custom landing page for users.
- Components:
 - Reports, Dashboards
 - Announcements
 - Custom LWC (e.g., Recent Videos)
- Can assign different layouts for different **profiles**.



5. Utility Bar

- Persistent footer in Lightning apps.
- Quick access tools for productivity.
- Examples for Video project:
 - Video Uploader (LWC)
 - Recent Videos
 - Alerts from Detection System
- Add via **App Manager** → **Utility Bar** → **Add Utility**.

App Settings

App Details & Branding

App Options

Utility Items (Desktop Only)

Navigation Items

User Profiles

Utility Items (Desktop Only)

Give your users quick access to productivity tools and add background utility items to your app.

Add Utility Item

Utility Bar Alignment ⓘ

Default

Omni-Channel

Notes

History

To Do List

List View

History

My Appointments

Report Chart

PROPERTIES

History

↑

↓

Remove

Utility Item Properties

Label ⓘ

History

Icon ⓘ

clock

Panel Width ⓘ

340

Panel Height ⓘ

480

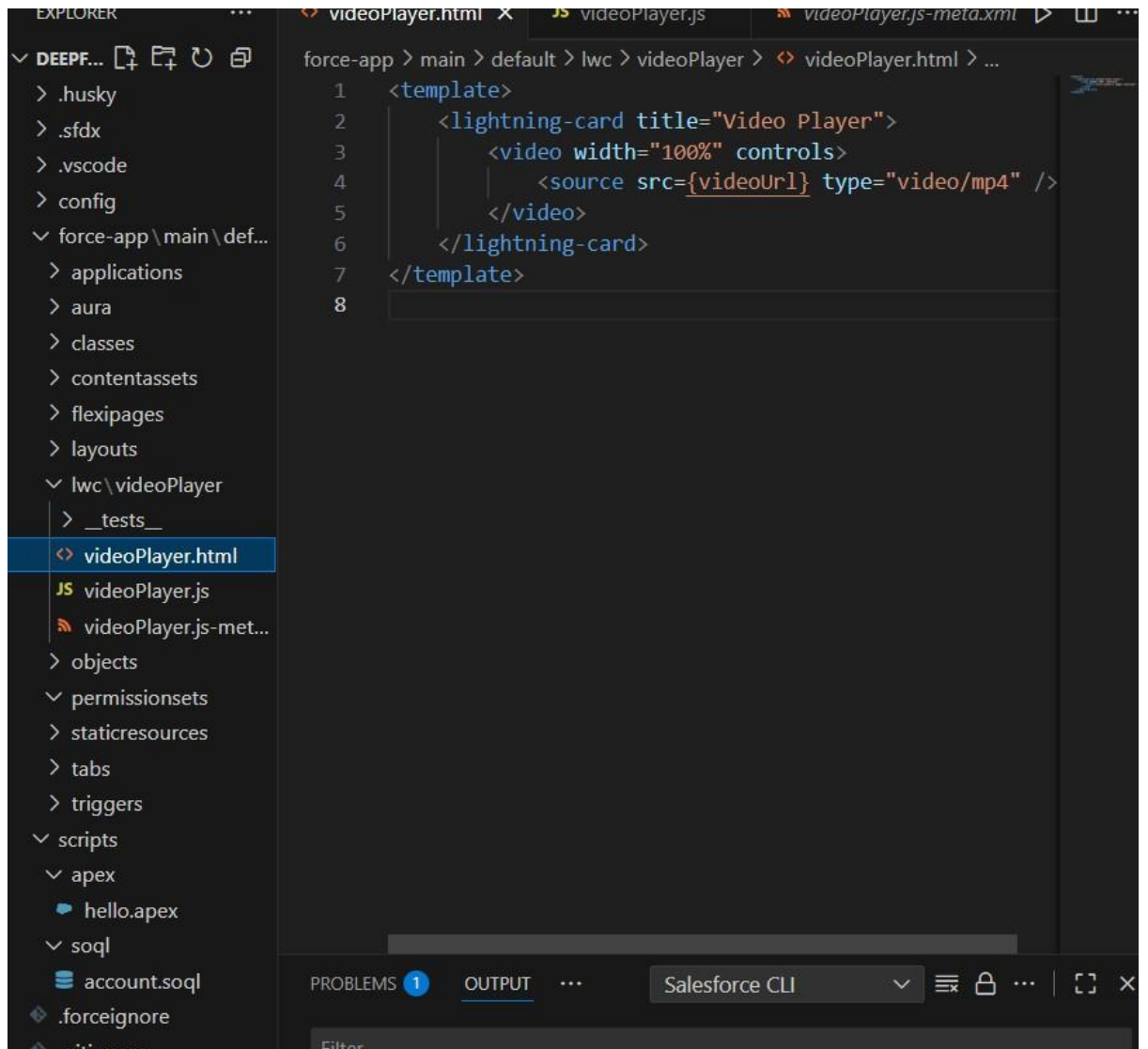
☒ Start automatically ⓘ

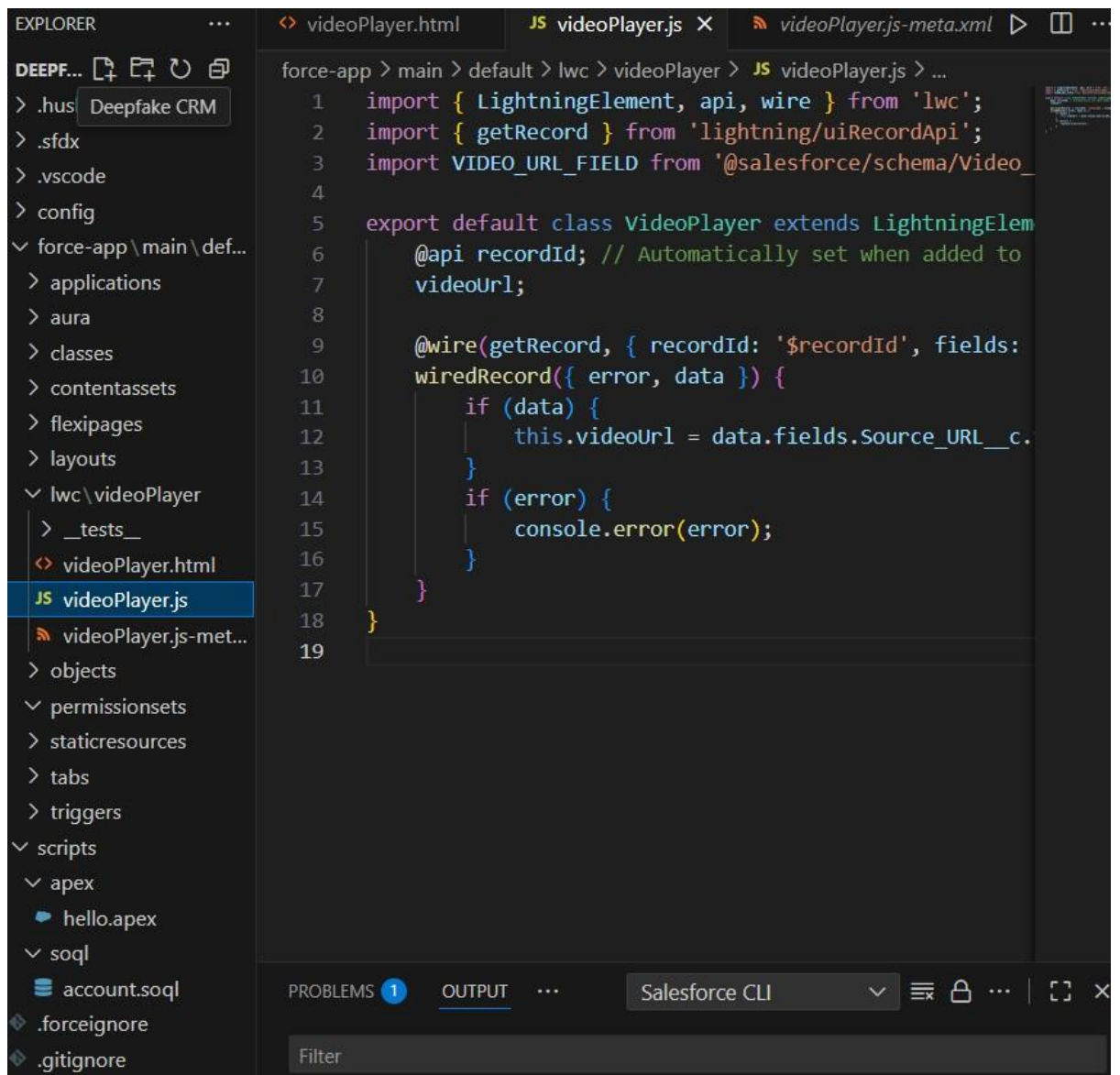
Cancel

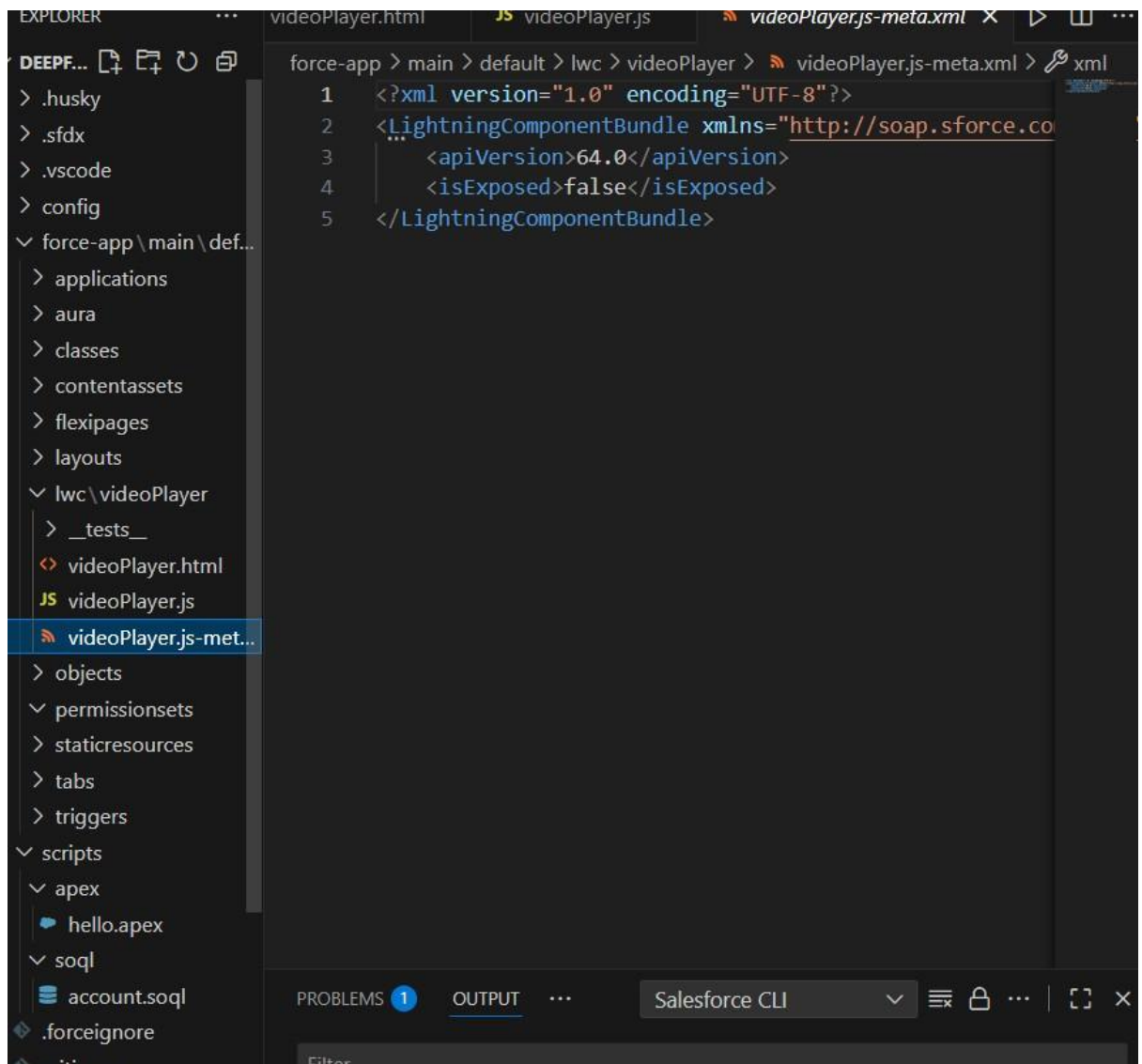
Save

6. Lightning Web Components (LWC)

- Modern framework based on **JavaScript ES6+**.
- Files:
 - .html → UI ◦ .js → Logic ◦ .css → Styling (optional)
- Can be added to Record Pages, App Pages, and Utility Bar.
- Example: Video Player, Video Alerts, Upload Widget.







7. Apex with LWC

- Apex handles **server-side logic**.
- **Read-only data:** Use **@AuraEnabled(cacheable=true)** with wire adapter.
- **User actions:** Use **imperative Apex calls** for buttons/events.
- Example:
 - Fetch Video details

- Update Video status ○ Analyze video
-

8. Events in LWC

- **Custom events:** Child → Parent communication.
 - **Pub/Sub / Application events:** Component → unrelated component communication.
 - Use case: Update Detection Status across multiple components when video is analyzed.
-

9. Wire Adapters

- Reactive data fetching.
 - Examples:
 - getRecord ○ getObjectInfo ○ getPicklistValues
 - Automatically updates component when data changes.
-

10. Imperative Apex Calls

- Explicit Apex call triggered by user action.
 - Example:
 - Analyze video ○ Approve video
 - Flexible control over when server-side code runs.
-

11. Navigation Service

- Use NavigationMixin in LWC.
- Programmatic navigation:
 - Standard pages (record, object list)
 - URLs
 - Lightning pages
- Example: Redirect user after saving or analyzing a video.
